R^3: A Real-Time Robust MU-MIMO Scheduler for O-RAN

Yubo Wu, *Student Member, IEEE*, Yi Shi, *Senior Member, IEEE*, Y. Thomas Hou, *Fellow, IEEE*, Wenjing Lou, *Fellow, IEEE*, Jeffrey H. Reed, *Life Fellow, IEEE*, and Luiz A. DaSilva, *Fellow, IEEE*

***Abstract*— Open Radio Access Network (O-RAN) offers a new paradigm for the design and deployment of future RANs. The unique architecture of O-RAN presents two main challenges when designing a scheduler. First, it is impractical to obtain accurate and full Channel State Information (CSI) due to estimation errors and limited bandwidth of the fronthaul link between Open Radio Unit (O-RU) and Open Distributed Unit (O-DU). Second, the large-scale processing at an O-DU introduces difficulties in meeting the stringent time requirement in O-RAN, especially in the real-time (RT) control loop. To address these challenges, we propose R^3—a real-time robust Multi-user, Multiple Input, Multiple Output (MU-MIMO) scheduler for O-RAN. R^3 serves as a comprehensive scheduling solution encompassing RB allocation, MCS selection, and beamforming calculation. Most notably, R^3 utilizes a limited number of CSI samples to offer probabilistic QoS guarantees. To meet the timing requirements of O-RAN, R^3 decomposes the scheduling problem into two distinct sub-problems and integrates them into separate control loops. Moreover, each sub-problem is designed with a parallel structure, utilizing a reduced search space, and implemented on a GPU platform to accelerate the computation time. Experimental results demonstrate that R^3 offers competitive throughput performance as the state-of-the-art while simultaneously fulfilling the QoS guarantees. Further, R^3 meets the timing requirements of various control loops in O-RAN over a wide range of operating conditions.**

***Index Terms*— CSI, MU-MIMO, real-time, O-RAN, scheduler.**

I. Introduction

THE push for a more open and flexible wireless network architecture has led to the development of O-RAN [1]. O-RAN offers a departure from the conventional proprietary RAN systems by promoting openness and interoperability [2], [3], [4], [5], [6]. At its core, O-RAN enables a new *"mix-and-match"* approach to RAN deployment and allows a carrier to choose the best hardware/software from different vendors. It breaks up the long-standing closed-RAN paradigm and moves the RAN market towards a more competitive and vibrant supply-chain ecosystem [7].

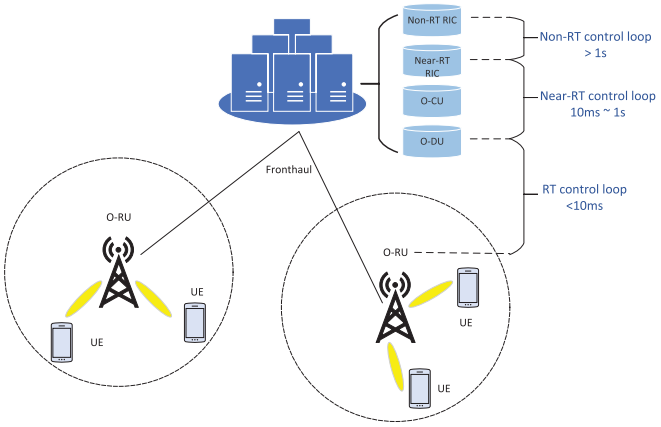


Fig. 1. An O-RAN reference architecture.

Figure 1 shows a reference architecture for O-RAN. In contrast to traditional RANs, O-RAN consists of five distinct components: Open Radio Unit (O-RU), Open Distributed Unit (O-DU), Open Central Unit (O-CU), near-real-time (near-RT) RAN Intelligent Controller (RIC) [8] and non-real-time (non-RT) RIC [9]. Each component operates at different time scales and belongs to three distinct control loops: RT (less than 10 ms), near-RT (10 ms to 1 s), and non-RT (over 1 s) control loops [10].

While O-RAN holds significant potential for opening up RANs' ecosystem, it also presents some significant challenges in designing its scheduler. First, just like traditional RANs, it is impractical to obtain perfect (full and accurate) Channel State Information (CSI) at an O-DU. Due to noise [11] and significant overhead [12], sending accurate and complete CSI from a UE to an O-RU is not feasible. Specifically, the use of a large number of pilot signals for full CSI transmission will adversely impact spectrum efficiency, particularly when multiple input, multiple output (MIMO) systems are involved. Second, unlike traditional RAN fronthaul link that carries user data bitstreams between the base station and the core network, the fronthaul link in O-RAN carries raw I/Q samples from the O-RU to the O-DU. Given the limited capacity of commercial fiber fronthaul links, such as Gigabit Ethernet

Manuscript received 21 January 2024; revised 2 June 2024 and 27 August 2024; accepted 28 August 2024. Date of publication 16 September 2024; date of current version 13 November 2024. This work was supported in part by NSF under Grant CNS-2312447, ONR MURI Grant N00014-19-1-2621, Virginia Commonwealth Cyber Initiative (CCI), and Virginia Tech Institute for Critical Technology and Applied Science (ICTAS). The associate editor coordinating the review of this article and approving it for publication was Z. Guan. *(Corresponding author: Y. Thomas Hou.)*

Yubo Wu, Y. Thomas Hou, and Jeffrey H. Reed are with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061 USA (e-mail: wuyubo@vt.edu; thou@vt.edu; reedjh@vt.edu).

Yi Shi and Luiz A. DaSilva are with the Commonwealth Cyber Initiative, Virginia Tech, Arlington, VA 22203 USA (e-mail: yshi@vt.edu; ldasilva@vt.edu).

Wenjing Lou is with the Department of Computer Science, Virginia Tech, Arlington, VA 22203 USA (e-mail: wjlou@vt.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2024.3456596.

Digital Object Identifier 10.1109/TWC.2024.3456596

TABLE I
NOTATIONS

Symbol	Definition
General Notation	
A_T	Number of transmit antennas at an O-RU
B	Total number of RBGs at the O-RU
K	Total number of UEs served by the O-RU
M	Maximum MCS level
Scheduling Notation	
x_k^b	Binary decision variable indicating whether or not RBG b is allocated to UE k
y_k^m	Binary decision variable indicating whether or not MCS level m is assigned to UE k
$\mathbf{h}_k^b$	An $A_T \times 1$ random CSI vector for UE k on RBG b
$P_{\max}$	Power budget on each RBG at the O-RU
$\mathbf{w}_k^b$	An $A_T \times 1$ decision variable vector for beamforming at O-RU w.r.t. UE k on RBG b
s_k^b	SINR at UE k on RBG b
r_k	Achievable data rate by O-RU w.r.t. UE k
Q_k	Required data rate by UE k
ϵ	Risk level
$\mathbf{X}$	A $K \times B$ matrix consisting of elements x_k^b
$\mathbf{Y}$	A $K \times (M+1)$ matrix consisting of elements y_k^m
$\mathbf{H}$	A $K \times B$ matrix consisting of elements $\mathbf{h}_k^b$
CSI Sample Notation	
N_s	Number of available CSI samples for each scheduling instance
γ	Overhead ratio of CSI transmission
$\hat{\mathbf{h}}_k^b$	An $A_T \times 1$ vector of CSI sample embedded with error for UE k on RBG b
$\mathbf{h}_k^{b*}$	An $A_T \times 1$ vector of error-free CSI sample for UE k on RBG b
$\hat{\mathbf{H}}_k^b$	A $1 \times N_s$ vector consisting of elements $\hat{\mathbf{h}}_k^b$
$\mathbb{P}_{\mathbf{h}_k^b}$	True distribution of $\mathbf{h}_k^b$
$\mathbb{P}_{\hat{\mathbf{h}}_k^b}$	Empirical distribution of $\hat{\mathbf{h}}_k^b$
$\mathbb{P}_{\mathbf{h}_k^{b*}}$	Empirical distribution of $\mathbf{h}_k^{b*}$
$\mathrm{d}(\mathbb{P}_{\mathbf{h}_k^b}, \mathbb{P}_{\hat{\mathbf{h}}_k^b})$	∞-Wasserstein distance between $\mathbb{P}_{\mathbf{h}_k^b}$ and $\mathbb{P}_{\hat{\mathbf{h}}_k^b}$
θ_k^b	Radius of the ambiguity set for UE k on RBG b

Passive Optical Network (GEPON) [13] (typically 10 Gbps), the fronthaul link is already strained by the I/Q samples. For example, in a 5G cell with 100 MHz bandwidth, the required bit rate for transmitting raw I/Q samples from each UE (with a 200 MHz sampling rate) is 200 MHz $\times$ 18 bits [14] $=$ 3.6 Gbps. This leaves little room to accommodate the full CSI overhead [15]. To address this challenge, the O-RAN specifications have devoted extensive coverage on overhead reduction techniques [15], [16]. Third, it is very challenging to meet the stringent timing requirement of O-RAN system. Complex tasks such as Resource Block (RB) allocation, Modulation and Coding Scheme (MCS) selection, and beamforming calculation should all be done at an O-DU. As we shall see in Section IV, the available time to complete these tasks in different control loops (tens of milliseconds for near-RT control loop and less than 10 ms for RT control loop) poses a major challenge to scheduler design.

There have been some active research efforts devoted to the design of Multi-user, Multiple Input, Multiple Output (MU-MIMO) schedulers. However, these approaches assume either perfect (complete and precise) CSI or accurate statistical characterizations of CSI, such as the mean and covariance of the CSI distribution. But in practice, obtaining any of such information is infeasible. Furthermore, there is little research that jointly optimizes the scheduling of RB allocation, MCS selection, and beamforming calculation while meeting the stringent timing requirement of O-RAN. More discussions on related work are given in Section II.

In this paper, we present R^3—a Real-time Robust MU-MIMO scheduler for O-RAN. The main contributions of R^3 are:

- Unlike most existing MU-MIMO schedulers that only address one or two components, R^3 stands out as a comprehensive solution that combines RB allocation, MCS selection, and beamforming calculation. Furthermore, R^3 is meticulously designed to specifically address the distinctive challenges posed by O-RAN, such as imperfect CSI and the need for real-time computation. None of the state-of-the-art schedulers can address these challenges effectively in a holistic manner.
- In contrast to existing schedulers that rely on perfect CSI or prior knowledge of channel statistics (model-based), R^3 addresses the challenge of imperfect (partial and inaccurate) CSI at O-DU in O-RAN by using a *data-driven* approach. More important, R^3 effectively addresses channel uncertainty by solely relying on *a small number* of CSI samples, while providing probabilistic QoS guarantees.
- To effectively meet the computational time requirements of O-RAN, R^3 employs a two-stage optimization approach, decomposing the scheduling problem into two distinct sub-problems within the near-RT and RT control loops in O-RAN. Further, each sub-problem is solved with a parallel structure, through a reduced search space, and making efficient use of the computational power offered by a GPU platform.
- Experimental results show that R^3 can offer high spectrum efficiency while successfully providing probabilistic QoS guarantees, even in the presence of imperfect CSI. R^3 also demonstrates robust performance across varying system parameters. R^3 is the only known scheduler that can meet the stringent time requirement of O-RAN with the aforementioned capabilities.

II. Related Work

In the domain of scheduler design for wireless communication systems, research bifurcates primarily into two distinct categories: schedulers based on perfect CSI and those predicated on imperfect CSI. This section provides a review of the current state-of-the-art within each category.

Schedulers based on perfect CSI (e.g., [17], [18], [19], [20], [21], [22], [23], [24]) operate under the assumption that both complete and accurate CSI are available. These schedulers leverage this idealized, error-free CSI to make scheduling decisions. While such designs are typically less complex and capable of achieving good performance, the realization of perfect CSI is hindered by practical issues (e.g., large overhead and estimation errors). The challenge of obtaining perfect CSI becomes particularly pronounced in O-RAN system, due to currently limited bandwidth of the fiber front-haul link.

Conversely, schedulers based on imperfect CSI (e.g., [25], [26], [27], [28], [29], [30]) operate under the premise that

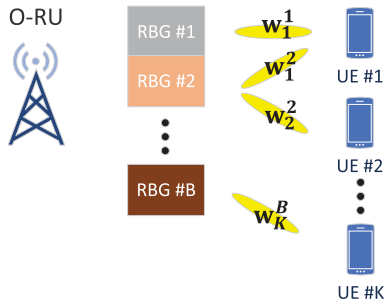


Fig. 2. An illustration of downlink MU-MIMO transmission at an O-RU.

CSI is dynamic and inherently uncertain. These schedulers often rely on the statistical characteristics of the CSI, aiming to deliver performance optimized over a time period. While these schedulers alleviate the need for perfect CSI–thereby reducing system overhead–they still require accurate estimates of the mean or covariance of the CSI distribution. However, the ephemeral and fluctuating nature of channel conditions renders these parameters elusive. The means and variances of the CSI distribution can only be estimated with a finite number of samples, introducing inherent errors. Consequently, the performance of the schedulers depends on the accuracy in the obtained statistical characteristics of the CSI.

Beyond the previously delineated limitations of contemporary scheduler designs, additional challenges are also prevalent. First, owing to the expansive search space inherent to scheduling decisions, a majority of algorithms elect to focus on a limited subset of decision factors, typically encompassing one or two elements such as RB allocation, MCS selection, or beamforming computation (e.g., [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32]). This approach, while simplifying the decision-making process, often falls short of attaining quality performance due to the absence of a comprehensive, joint optimization strategy.

Secondly, the intricate nature and extensive scope of a scheduler often lead to complex algorithms (e.g., [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [32]). These algorithms are usually plagued by excessive computation times, which cannot meet the stringent real-time requirement in O-RAN, especially the RT control loop.

In summary, none of the existing MU-MIMO schedulers can effectively address the imperfect CSI challenge in O-RAN while achieving high performance and meeting the time requirements of O-RAN.

III. System and Mathematical Models

Consider a downlink (DL) scheduling problem within O-RU service area in the O-RAN architecture (see Fig. 2). Denote A_T as the number of antennas at an O-RU and we assume the number of antennas at each UE is 1. Denote B as the number of available RB groups (RBGs).[1] If an RBG $b = 1, \cdots, B$ is assigned to a UE $k = 1, \cdots, K$, there is an $A_T \times 1$ beamforming vector $\mathbf{w}_k^b$ to be determined by the O-RU to optimize system performance.

[1]We use RGB as the minimum resolution in resource allocation instead of RB for generality and time efficiency [42].

A. CSI

As depicted in Fig. 1, the CSI is initially conveyed from UEs to O-RU via the wireless channel, and subsequently transmitted from the O-RU to O-DU through an optical fronthaul link. Obtaining perfect (full and accurate) CSI is impossible due to several practical reasons. First, between UEs and the O-RU, the required bit rate for transmitting full CSI can be up to Gbps. For instance, considering a typical cell with 40 users, 120 RBs, and 8 antennas, the bit rate needed for transmitting full CSI can be calculated as $40 \times 120 \times 8 \times 32 = 1.22$ Gbps. Such a large overhead from the UEs simply cannot be supported. Second, the CSI feedback process is susceptible to errors such as channel noise and hardware noise. These errors will adversely impact the accuracy of the CSI feedback. Third, for uplink transmission between an O-RU and the O-DU, the O-RU needs to sample the received I/Q signals, quantize the samples, and then forward them to the O-DU. For a UE operating with several MHz bandwidth, the bandwidth used for transmitting I/Q signals can escalate to several Gbps [15]. But the current commercial fiber links typically only have ~10 Gbps capacity. So there is simply no room to carry full CSI, let alone to say that the same uplink fiber still needs to carry user data, which is supposed to be the dominant traffic component.

Given that perfect (full and accurate) CSI is not available, for each UE k, the CSI for RBG b can be represented by an $A_T \times 1$ vector of random variables, which we denote as $\mathbf{h}_k^b$. In this paper, we do not assume any prior knowledge of the distributions of the random variables. Instead, we will only rely on *a small number* of CSI samples, which represent imperfect (partial and inaccurate) CSI. How this can be done will be explored in Section VII.

B. RB Allocation

Denote x_k^b as a binary decision variable with the following definition:

$$x_k^b = \begin{cases} 1, & \text{if RBG } b \text{ is allocated to UE } k, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Under MU-MIMO, one RBG can be allocated to multiple (up to A_T) UEs. This is a fundamental differentiating feature from SU-MIMO. So we have:

$$\sum_{k=1}^{K} x_k^b \le A_T. \qquad (b = 1 \cdots B) \tag{2}$$

C. MCS Assignment

Denote y_k^m as a binary decision variable with the following definition:

$$y_k^m = \begin{cases} 1, & \text{if MCS level } m \text{ is assigned to UE } k, \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

where $m \in [0, M]$ and $M = 28$ per 5G standard [33].

Since only one MCS can be assigned to a UE (which will be used on all RBGs allocated to this UE), we have:

$$\sum_{m=0}^{M} y_k^m = 1 \qquad (k = 1 \cdots K). \tag{4}$$

Constraint (4) is a distinct feature for MCS assignment for a UE.

D. O-RU Transmit Power

For the beamforming vector $\mathbf{w}_k^b$, we have the following constraint:

$$\sum_{k=1}^{K} ||\mathbf{w}_k^b||_2^2 \leq P_{\max} \quad (b = 1 \cdots B), \tag{5a}$$

$$||\mathbf{w}_k^b||_2^2 \leq x_k^b P_{\max} \quad (b = 1 \cdots B, k = 1 \cdots K), \tag{5b}$$

where $||\cdot||_2$ denotes L_2 norm and $P_{\max}$ denotes the maximum transmit power on each RBG at an O-RU. If RBG b is not assigned to UE k, then $||\mathbf{w}_k^b||_2^2 = 0$ (i.e., the beamforming vector $\mathbf{w}_k^b$ is 0).

E. SINR and Data Rate

Denote N_k^b as the noise power at UE k on RBG b. The SINR at UE k on RBG b, denoted as s_k^b, can be calculated as follows:

$$s_k^b = \frac{|(\mathbf{w}_k^b)^\dagger \mathbf{h}_k^b|^2}{\sum_{i=1}^{K, i \neq k} |(\mathbf{w}_i^b)^\dagger \mathbf{h}_k^b|^2 + N_k^b}, \tag{6}$$

where $(\cdot)^\dagger$ denotes the conjugate transpose.

Based on (6), the achievable data rate of UE k on RBG b with MCS $y_k^m = 1$, denoted as $r_k^{b,m}$, can be calculated as follows (see Table 5.1.3.1-1 in [33]):

$$r_k^{b,m} = \begin{cases} R_m, & \text{if } \ s_k^b \geq S_m, \\ 0, & \text{otherwise}, \end{cases} \tag{7}$$

where R_m is the achievable data rate of MCS level m and S_m is the SINR threshold of MCS level m, $m \in [0, M]$. For a given m value, if $s_k^b \geq S_m$, then $r_k^{b,m} = R_m$ regardless how large the difference between s_k^b and S_m. On the other hand, if $s_k^b < S_m$, then $r_k^{b,m} = 0$ even though s_k^b may be close to S_m. So a judicious choice of m is critical to maximizing throughput, especially when multiple RBGs with different SINRs are allocated to the same UE k.

Based on (4) and (7), the achievable data rate of UE k on RBG b, denoted as r_k^b, is as follows:

$$r_k^b = \sum_{m=0}^{M} y_k^m r_k^{b,m}. \tag{8}$$

Finally, the total data rate of UE k over all RBGs allocated to it, denoted as r_k, is:

$$r_k = \sum_{b=1}^{B} r_k^b \tag{9}$$

F. Data Rate Guarantee

Given the imperfect CSI, it is impossible to offer a hard (deterministic) guarantee for data rate r_k. Instead, we propose to offer a probabilistic guarantee via *chance constraints*.

Denote Q_k as the required QoS data rate from UE k (in bps) and ϵ as the *risk level* of violating the probabilistic (chance) constraint. We can put forth the following chance constraints:

$$P\Big\{ r_k \geq Q_k \Big\} \geq 1 - \epsilon \ \ (k = 1 \cdots K) \tag{10}$$

where $P\{\cdot\}$ denotes the probability measure function. The QoS chance constraint aims to guarantee that the system offers a QoS data rate requirement for each UE with a probability of at least $1 - \epsilon$, where ϵ is a small value close to 0. This kind of guarantee in chance constraints accounts for occasional deviations from the desired QoS levels while maintaining high overall service quality.

IV. Problem Formulation

A. Objective Function

Many objective functions can be considered. In this paper, we set the objective function to maximize the overall system throughput (i.e., the sum of throughput from all UEs).

Denote $r(t)$ as the system throughput at time t. Then our objective function is to

$$\max \ \mathbb{E}_{\mathbf{H}} \Big[\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} r(t) \Big] \tag{11}$$

where $\mathbf{H} = [\mathbf{h}_k^b]_{K \times B}$. Note that in (11), we take the expectation due to the random nature of $\mathbf{H}$. By moving the expectation function inside the sum, the objective function becomes: $\max \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_{\mathbf{H}} \Big[r(t) \Big]$. Since scheduling decisions at each time t (as $T \to \infty$) are independent, to maximize $\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_{\mathbf{H}} \Big[r(t) \Big]$, it is sufficient to

$$\max \ \mathbb{E}_{\mathbf{H}} \Big[r(t) \Big] \tag{12}$$

for each scheduling time instance (or TTI) t. We can omit time dependence (t) when there is no confusion. Since $r = \sum_{k=1}^{K} r_k$, (12) is equivalent to:

$$\max \ \mathbb{E}_{\mathbf{H}} \Big[\sum_{k=1}^{K} r_k \Big], \tag{13}$$

which is our objective function.

B. Problem Formulation

Putting together the objective function and constraints, we have the following stochastic programming problem with chance constraints [34]:

$$\textbf{OPT-S} \quad \max \ \mathbb{E}_{\mathbf{H}} \Big[\sum_{k=1}^{K} r_k \Big]$$

s.t. RB allocation constraint: (2);

MCS selection constraint (4);
Power constraint (5);
SINR calculation (6);
Data rate calculation (7)(8)(9);
QoS chance constraint (10);
$x_k^b \in \{0,1\}; y_k^m \in \{0,1\}; \mathbf{w}_k^b \in \mathbb{C}$,

where x_k^b, y_k^m and $\mathbf{w_k^b}$ are decision variables, B, K, M, A_T, $P_{\max}$, $[R_0, \cdots, R_m]$, $[S_0, \cdots, S_m]$, ϵ and Q_k are constants, $\mathbf{h}_k^b$ is a vector of random variables, s_k^b and r_k are both intermediate variables.

A scheduler in O-RAN should do the scheduling (maximizing the throughput while offering the QoS guarantee without knowing the actual CSI) by solving OPT-S in real-time. Problem OPT-S is hard to solve for at least four reasons. First, the coupling of all three decision variables presents a significant challenge in finding a solution. Second, the inclusion of the random variable $\mathbf{h}_k^b$ and probabilistic (chance) constraints introduce uncertainty to the optimization problem, thereby elevating its complexity. Third, the search space for each decision variable is prohibitively large. The joint RB allocation and MCS selection problem is already NP-hard [22]. Finally, scheduling at an O-RU has stringent real-time requirements (on the order of ∼ms). This means that a solver to OPT-S must meet this RT requirement.

V. R^3: Main Ideas

To solve OPT-S in real-time, we propose R^3—a real-time robust MU-MIMO scheduler in the O-RAN architecture. The main ideas in R^3 consist of the following four elements.

First, to mitigate the complexity of OPT-S, R^3 exploits the two different time scales associated with Near-RT and RT control loops in O-RAN to decouple it into two sub-problems: OPT-S1 (for RB allocation and MCS assignment) and OPT-S2 (for beamforming calculation). This decoupling aims to solve the decision variables x_k^b and y_k^m through OPT-S1 before solving $\mathbf{w}_k^b$ in OPT-S2. Since the beaming variables $\mathbf{w}_k^b$ may still be present in OPT-S1, we can initialize them with some well-known (provably good) solutions and then re-calibrate them when we solve OPT-S2.

Second, R^3 addresses the issue of imperfect CSI by using a small number of CSI samples, which substantially reduces CSI overhead. Based on the small number of CSI samples, R^3 removes our problems' dependency on the random variable $\mathbf{h}_k^b$. Specifically, R^3 reformulates OPT-S1 and OPT-S2 into two deterministic formulations OPT-D1 and OPT-D2, respectively, by utilizing sample average approximation (SAA) and ∞-Wasserstein distance.

Third, R^3 solves OPT-D1 in near-RT control loop and then OPT-D2 in the RT control loop. For each problem, we propose to reduce the search space for the final high-performing solution. The goal is to strike a balance between solution quality and computational efficiency. In particular, for OPT-D1, the search space for RB allocation decision variable x_k^b is reduced by exploiting the properties of MU-MIMO; The search space for MCS assignment decision variable y_k^m is reduced by considering the QoS requirements of each user. For OPT-D2, we first form a signal basis using the most promising beamforming vectors. Then the signal basis is used to generate a spanning space (a reduced search space) through linear combinations.

Fourth, based on the reduced search space for each problem, R^3 employs parallel computing (using a GPU platform) to accelerate computation time for finding a high-performing solution. Specifically, for OPT-D1 or OPT-D2, R^3 incorporates a carefully designed multi-layer structure, with each layer customized for parallel computation. At each layer, a large number of independent sub-problems are executed in parallel in GPU cores. To minimize computation time with a given GPU platform, R^3 meticulously engineers efficient utilization of given GPU resources, including kernel, block, thread, and shared memory.

In the subsequent sections, we elaborate on these key design elements in R^3.

VI. Problem Decomposition

For the problem formulation in OPT-S, we recognize that in practice, RB allocation and MCS assignment are typically done before setting the beamforming vector. This motivates us to explore decomposing OPT-S into two independent sub-problems. Specifically, we observe that RB allocation and MCS selection mainly depend on channel gain, which remains relatively stable on the time scale of ∼10 ms. So RB allocation and MCS selection can be effectively performed within the *near-RT* control loop in O-RAN. In contrast, beamforming vector calculation depends on both channel gain and phase, which can change at ∼1 ms time scale. So it is most suitable to perform beamforming vector calculation in the *RT* control loop.

Based on the above discussion, we replace OPT-S with two sub-problems OPT-S1 and OPT-S2 as follows:

$$\textbf{OPT-S1} \quad \max \ \mathbb{E}_{\mathbf{H}}\left[\sum_{k=1}^{K} r_k\right]$$
s.t. RB allocation constraint: (2);
MCS selection constraint (4);
Constraints: (6)(7)(8)(9);
$x_k^b \in \{0,1\}; y_k^m \in \{0,1\}$,

and

$$\textbf{OPT-S2} \quad \max \ \mathbb{E}_{\mathbf{H}}\left[\sum_{k=1}^{K} r_k\right]$$
s.t. Power constraint (5);
QoS chance constraint (10);
Constraints: (6)(7)(8)(9);
$\mathbf{w}_k^b \in \mathbb{C}$.

Note that this decomposition is not perfect, as beamforming variables $\mathbf{w_k^b}$ still appear in OPT-S1 through constraints (6). To address this issue, we propose the following solution procedure (see Fig. 3).

First, for OPT-S1, we will set the beamforming vectors $[\mathbf{w}_k^b]_{K\times B}$ to be constant so that we can find $[x_k^b]_{K\times B}$ and

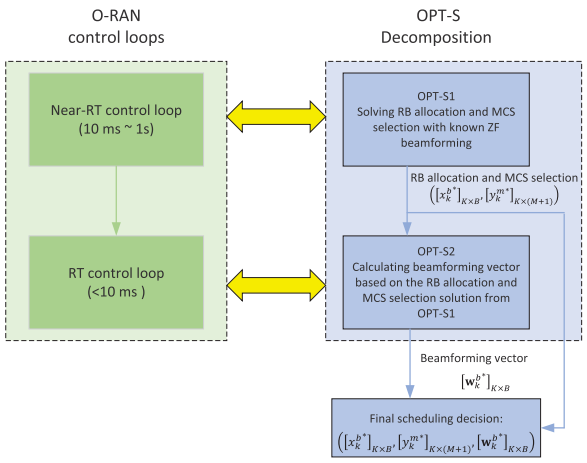


Fig. 3. A mapping of our problem decomposition (OPT-S1 and OPT-S2) into O-RAN's near-RT and RT control loops.

$[y_k^m]_{K\times(M+1)}$. This can be done by using Zero Forcing (ZF) beamforming [36] for $[\mathbf{w}_k^b]_{K\times B}$ as a starting point. ZF is deemed near-optimal for maximizing total throughput.[2] Subsequently, with the output from the solution to OPT-S1, i.e., $[x_k^{b\,*}]_{K\times B}$ and $[y_k^{m*}]_{K\times(M+1)}$, we can solve the beamforming vectors $[\mathbf{w}_k^b]_{K\times B}$ in OPT-S2. This new objective value will further improve the one that we achieved earlier in OPT-S1.

The above two-step procedure is not entirely equivalent to solving OPT-S. But it is a sound heuristic that will offer a highly competitive solution. More importantly, it effectively utilizes the two control loops of different time scales in O-RAN to address the three sets of decision variables.

However, after decomposition, OPT-S1 and OPT-S2 remain stochastic programming problems (due to random variables $\mathbf{h}_k^b$'s). In this paper, we do not assume any prior knowledge of these random variables' distributions. Instead, we will rely on a small number of CSI samples to address these random variables.

VII. From Random Variables to Limited CSI Samples

R^3 employs a small number of CSI samples to represent random variable $\mathbf{h}_k^b$ (see Fig. 4). Denote $\hat{\mathbf{H}}_k^b$ as the small set of CSI samples of UE k on RBG b, i.e.,

$$\hat{\mathbf{H}}_k^b = \left[\,\hat{\mathbf{h}}_k^b(1)\ \hat{\mathbf{h}}_k^b(2)\ \cdots\ \hat{\mathbf{h}}_k^b(N_s)\,\right] \tag{14}$$

where $\hat{\mathbf{h}}_k^b(j)$ $(j = 1\cdots N_s)$ is an $A_T \times 1$ vector representing j-th received CSI sample from UE k on RBG b, N_s is the number of CSI samples that we employ over a time window of L TTIs. Note that the matrix $\hat{\mathbf{H}}_k^b$ only contains a subset of all possible CSI samples within the time window L (marked in blue in Fig. 4). Within window L, the CSI experiences changes in each TTI. However, its distribution is considered stable, reflecting the actual channel behavior, including path loss and fast fading, observed in real-world conditions.

[2]Another well-known beamforming method such as MMSE [37] does not provide substantial benefits in our context, as the single receiving antenna already prevents noise amplification at the receiver side.

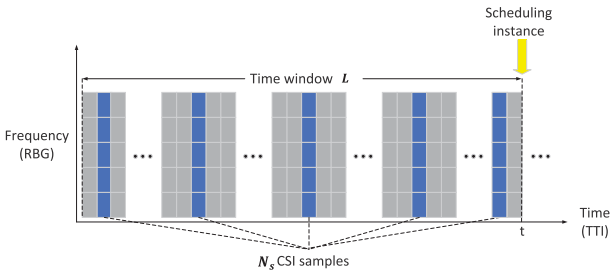


Fig. 4. Scheduling at time instance t based on a small number of CSI samples (marked in blue) in the past time window L.

The determination of parameter L will be further elaborated in Section X. Denote $\gamma = \frac{N_s}{L}$ as an overhead ratio that quantifies the percentage of CSI samples employed. The range of γ is $(0, 1]$. Clearly, the larger the γ is, the higher the overhead. We will also investigate the impact of γ setting on the solution performance in Section X.

Since the reformulation of OPT-S2 is more complex than OPT-S1, we will discuss it first. In OPT-S2, the random variable $\mathbf{h}_k^b$ appears in both the objective function and the QoS chance constraints (10). So we need to reformulate both.

For the objective function, with $\hat{\mathbf{H}}_k^b$, it can be reformulated as a deterministic form by using sample average approximation (SAA) [38] as follows:

$$\max\ \frac{1}{N_s}\sum_{j=1}^{N_s}\left\{\sum_{k=1}^{K} r_k \middle| \mathbf{h}_k^b = \hat{\mathbf{h}}_k^b(j), b = 1\cdots B\right\}. \tag{15}$$

For QoS constraints (10), with $\hat{\mathbf{H}}_k^b$, it can be reformulated as a deterministic form as follows:

$$\sum_{j=1}^{N_s}\mathbb{I}\left\{r_k \geq Q_k \middle| \mathbf{h}_k^b = \hat{\mathbf{h}}_k^b(j), b = 1\cdots B\right\} \geq N_s(1-\epsilon) \quad (k = 1\cdots K) \tag{16}$$

where $\mathbb{I}(\cdot)$ is a binary indicator function, returning 1 if the argument is true and 0 otherwise. For the j-th term in the summation on the LHS of (16), r_k is calculated using $\hat{\mathbf{h}}_k^b(j)$ based on (6)(7)(8)(9). This reformulation implies that to meet the QoS constraints (10) for the random variable $\mathbf{h}_k^b$, sample-based QoS constraints (16) must be satisfied for at least $N_s(1-\epsilon)$ CSI samples.

There are two important premises of using (16) to replace (10): i) a sufficient number of CSI samples; and ii) accurate CSI samples. Getting accurate CSI samples is not possible due to the estimation errors in $\hat{\mathbf{H}}_k^b$. To address this issue, we resort to ∞-*Wasserstein distance* [39], [40].

Denote $\mathbf{h}_k^{b*}$ as the discrete random variable for N_s data samples free of any estimation errors. Denote $\mathbb{P}_{\mathbf{h}_k^{b*}}$ as the empirical distribution of $\mathbf{h}_k^{b*}$ based on N_s data samples.[3] Denote $\mathbb{P}_{\hat{\mathbf{h}}_k^b}$ as the empirical distribution for $\hat{\mathbf{h}}_k^b$, which is a random variable for the N_s data samples (embedding estimation errors). We use the ∞-*Wasserstein distance* to characterize the *distance* between $\mathbb{P}_{\mathbf{h}_k^{b*}}$ and $\mathbb{P}_{\hat{\mathbf{h}}_k^b}$. Denote $\mathrm{d}(\mathbb{P}_{\mathbf{h}_k^{b*}}, \mathbb{P}_{\hat{\mathbf{h}}_k^b})$ as

[3]When $N_s \to \infty$, $\mathbf{h}_k^{b*} \to \mathbf{h}_k^b$ and $\mathbb{P}_{\mathbf{h}_k^{b*}} \to \mathbb{P}_{\mathbf{h}_k^b}$.

the ∞-Wasserstein distance between $\mathbb{P}_{\mathbf{h}_k^{b*}}$ and $\mathbb{P}_{\hat{\mathbf{h}}_k^b}$. We have:

$$\mathrm{d}(\mathbb{P}_{\mathbf{h}_k^{b*}}, \mathbb{P}_{\hat{\mathbf{h}}_k^b}) = \inf_{\mathbb{Q}\in\mathcal{Q}} \left\{ \sup_{\mathbb{Q}} ||\mathbf{h}_k^{b*} - \hat{\mathbf{h}}_k^b||_2 \right\}, \quad (17)$$

where $\mathcal{Q}$ stands for the set of all possible joint distributions of $\mathbf{h}_k^{b*}$ and $\hat{\mathbf{h}}_k^b$. Denote θ_k^b as the radius (a non-negative number) and $\mathcal{P}^{\theta_k^b}_{\mathbb{P}_{\hat{\mathbf{h}}_k^b}}$ as a set of all possible distributions $\mathbb{P}$ whose distance to the empirical distribution $\mathbb{P}_{\hat{\mathbf{h}}_k^b}$ is bounded by θ_k^b, i.e.,

$$\mathcal{P}^{\theta_k^b}_{\mathbb{P}_{\hat{\mathbf{h}}_k^b}} = \left\{ \mathbb{P} : \mathrm{d}(\mathbb{P}, \mathbb{P}_{\hat{\mathbf{h}}_k^b}) \le \theta_k^b \right\}. \quad (18)$$

If θ_k^b is chosen properly, $\mathbb{P}_{\mathbf{h}_k^{b*}}$ can fall within $\mathcal{P}^{\theta_k^b}_{\mathbb{P}_{\hat{\mathbf{h}}_k^b}}$ almost surely, i.e., $\mathbb{P}_{\mathbf{h}_k^{b*}} \in \mathcal{P}^{\theta_k^b}_{\mathbb{P}_{\hat{\mathbf{h}}_k^b}}$. So the question is: What value should we choose for θ_k^b?

Note that by using L_2 norm in (17), θ_k^b only depends on the maximum noise power (which is a given constant or can be easily found through offline learning) and the change of distribution's envelope (which depends on path loss). Since the latter is negligible within $\sim$10 ms time scale, we can confidently set θ_k^b as the square root of the maximum noise power (which includes all environmental noise).

Our goal is to ensure (16) is satisfied almost surely with $\mathbb{P}_{\mathbf{h}_k^{b*}}$. Since $\mathbb{P}_{\mathbf{h}_k^{b*}} \in \mathcal{P}^{\theta_k^b}_{\mathbb{P}_{\hat{\mathbf{h}}_k^b}}$, if we can ensure that (16) is satisfied for all distributions in $\mathcal{P}^{\theta_k^b}_{\mathbb{P}_{\hat{\mathbf{h}}_k^b}}$, then our goal can be achieved. This means the QoS inequality (for the j-th term in (16)) should be checked for all possible vectors $\mathbf{c}_k^b(j)$ as follows:

$$||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b, \quad (b = 1 \cdots B). \quad (19)$$

(19) holds because, the L_2 norm distance between two samples equals the ∞-Wasserstein distance between the two distributions. It is noted that although the scheduling solution is based on $\mathbb{P}_{\mathbf{h}_k^{b*}}$, the validation (QoS guarantee) will be conducted based on the original OPT with true unknown distribution $\mathbb{P}_{\mathbf{h}_k^b}$.

With (19), we can replace (16) with another deterministic form as:

$$\sum_{j=1}^{N_s} \mathbb{I}\Big\{ r_k \ge Q_k \Big| \mathbf{h}_k^b = \mathbf{c}_k^b(j),\ ||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b,\ b = 1 \cdots B \Big\} \ge N_s(1-\epsilon) \qquad (k = 1 \cdots K). \quad (20)$$

So we have a deterministic (D) reformulation of OPT-S2, denoted as OPT-D2, as follows:

OPT-D2

$$\max \ \frac{1}{N_s} \sum_{j=1}^{N_s} \left\{ \sum_{k=1}^{K} r_k \Big| \mathbf{h}_k^b = \hat{\mathbf{h}}_k^b(j), b = 1 \cdots B \ , k = \cdots K \right\}$$

s.t. Power constraint: (5);
QoS constraint: (20);
Constraints: (6)(7)(8)(9);
$\mathbf{w}_k^b \in \mathbb{C}$.

Although OPT-D2 is a deterministic optimization problem, the search space for $\mathbf{w}_k^b$ is still extremely large. Further, the QoS constraints (20) need to be checked for an infinite number of vectors per (19). In Section IX, we will address these issues.

For OPT-S1, random variable $\mathbf{h}_k^b$ only shows up in the objective function. So its deterministic reformulation (based on limited CSI samples) is:

OPT-D1

$$\max \ \frac{1}{N_s} \sum_{j=1}^{N_s} \left\{ \sum_{k=1}^{K} r_k \Big| \mathbf{h}_k^b = \hat{\mathbf{h}}_k^b(j), b = 1 \cdots B, k = 1 \cdots K \right\}$$

s.t. RB allocation constraint: (2);
MCS selection constraint (4);
Constraints: (6)(7)(8)(9);
$x_k^b \in \{0,1\}; y_k^m \in \{0,1\}$.

OPT-D1 is a deterministic optimization problem with extremely large search space for x_k^b and y_k^m. We will address these issues in Section VIII.

VIII. A Design of the Near-RT Control Loop

A. Basic Idea

For the near-RT control loop, R^3 focuses on solving OPT-D1 to find $([x_k^{b\,*}]_{K\times B}, [y_k^{m*}]_{K\times(M+1)})$. Since results obtained from solving OPT-D1 will be subsequently used in OPT-D2, it is necessary to ensure $([x_k^{b\,*}]_{K\times B}, [y_k^{m*}]_{K\times(M+1)})$ satisfy QoS constraints.

Problem OPT-D1 is NP-hard [21] and its solution space remains very large. For example, consider a system with 40 UEs, 10 RBGs, 8 antennas, and 28 MCS levels. The size of the original search space of OPT-D1 is $(|28|)^{|40|}\left[\binom{40}{1} + \cdots + \binom{40}{8}\right]^{10} \sim 10^{120}$ which is prohibitively large. To find a good solution to OPT-D1 within the specified time constraint of tens of milliseconds, R^3 employs the following steps.

First, R^3 reduces the search space for x_k^b by exploiting the properties of MU-MIMO. In the MU-MIMO system, it is advantageous to allocate RBG to the UEs that have high channel quality and low channel correlation. By exploiting this property, R^3 prunes the search space, resulting in a substantial reduction in the number of candidate UEs for each RBG.

Second, with the reduced space for RB allocation, R^3 further reduces the search space for y_k^m by narrowing its upper and lower bounds based on channel quality and the QoS constraints. Consequently, the number of candidate MCS levels for each UE is also significantly decreased.

Third, for each candidate decision pair $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$, R^3 performs a feasibility check on the QoS constraints. This step ensures that the selected RB allocation and MCS levels are feasible in terms of QoS requirements. If infeasible, the corresponding $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$ pair will be dropped.

Finally, R^3 computes and compares the objective values of all feasible solutions within the reduced search space. The solution that yields the highest objective value is selected as the final solution for RB allocation and MCS selection scheduling decision for OPT-D1.

B. Some Details

In the following, we describe these steps in detail.

1) Step 1. Search Space Reduction for x_k^b: We employ the following two fundamental properties in MU-MIMO to reduce search space for x_k^b. First, it is advantageous to allocate each RBG to the UE that exhibits good channel quality on that RBG. When a UE experiences poor channel quality on a specific RBG, assigning this RBG to the UE would not achieve a good data rate and would only introduce interference to other UEs sharing the same RBG. Second, it is preferable to allocate RBGs to UEs with minimal channel correlation. When there is a high channel correlation between different UEs on the same RBG, the spatial separability of the UEs' channels decreases and it becomes harder to separate data to different UEs, diminishing the benefits of MU-MIMO in increasing capacity.

Based on the first property, on RBG b, we will narrow down the set of suitable UEs from all K to a subset of K. To do this, we calculate the channel quality of UE k based on the average channel gain among all CSI samples as $||\bar{\mathbf{h}}_k^b||_2^2 = \frac{\sum_{j=1}^{N_s} ||\hat{\mathbf{h}}_k^b(j)||_2^2}{N_s}$. Suppose BW is the bandwidth of one RBG, then the maximum achievable data rate of UE k on RBG b can be estimated as follows:

$$q_k^b = \text{BW} \log\left(1 + \frac{P_{\max}||\bar{\mathbf{h}}_k^b||_2^2}{N_k^b}\right) \tag{21}$$

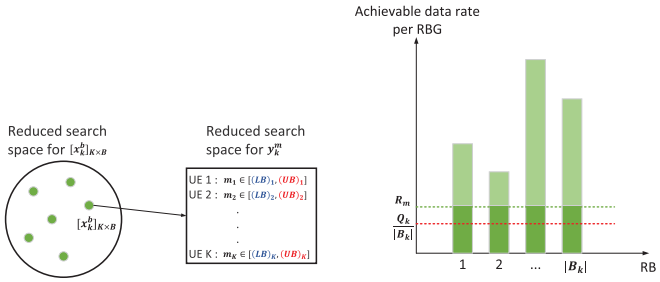
After computing all q_k^b's, the largest $G_{\text{ rate}}$ q_k^b's will be chosen as the candidate UEs on RBG b. It is noted that due to fast fading, each UE might encounter varying channel gains across different RBGs. This means if one UE is not selected as the candidate on a specific RBG, it can still be selected as a candidate on other RBGs. In other words, it is unlikely that a UE will not be selected by all RBGs in their sets. After this step, the number of candidate UEs on each RBG is reduced from K to $G_{\text{ rate}}$ $(< K)$.

Based on the second property, we further reduce the number of candidate UEs on each RBG by considering the correlation factors. For RBG b, we first calculate the channel correlations between every two UEs in the reduced UE search space. For UE k_1 and k_2 on RBG b, their correlation factor can be calculated as follows:

$$\frac{|\sum_{j=1}^{N_s}(\hat{\mathbf{h}}_{k_1}^b(j))^\dagger \hat{\mathbf{h}}_{k_2}^b(j)|}{N_s}. \tag{22}$$

For all pair-wise UEs in the $G_{\text{ rate}}$ UEs, we calculate the correlation factor for the pairs. Then we will choose the UE pairs starting with the smallest correlation factors (then the second smallest and so forth) until the number of distinct UEs reduces from $G_{\text{ rate}}$ to $G_{\text{ corr}}$ $(< G_{\text{ rate}})$. After this step, the number of candidate UEs for each RBG is $G_{\text{ corr}}$. G_{rate} and G_{corr} are chosen based on the computing capability of the GPU hardware. Random sampling is also implemented when the size of the search space exceeds the parallel computing capacity of the GPU hardware [35].

2) Step 2. Search Space Reduction for y_k^m: Step 1 reduces the search space of UEs on each RBG, which results in a reduction of search space for all x_k^b's, i.e., $[x_k^b]_{K\times B}$. In Step 2, we move on to reduce the search space for MCS selection w.r.t each UE k in an RB allocation matrix $[x_k^b]_{K\times B}$ in the reduced search space. That is, for each RB allocation matrix $[x_k^b]_{K\times B}$, we will find an upper bound and lower bound for MCS w.r.t. each UE k (see Fig. 5(a)).



(a) Search space reduction for MCS (b) MCS lower bound

Fig. 5. An illustration of search space reduction for y_k^m.

To set an upper bound for MCS w.r.t UE k in an RB allocation $[x_k^b]_{K\times B}$, we will find an upper bound for MCS level that can be used on all the RBGs that are allocated to this UE. Denote $\mathcal{B}_k$ as the set of candidate RBGs that can be allocated to UE k after Step 1. Since the SINR of UE k on RBG b $(b \in \mathcal{B}_k)$ satisfies:

$$s_k^b = \frac{|(\mathbf{w}_k^b)^\dagger \mathbf{h}_k^b|^2}{\sum_{i=1}^{K, i\neq k} |(\mathbf{w}_i^b)^\dagger \mathbf{h}_k^b|^2 + N_k^b} \leq \frac{P_{\max}||\bar{\mathbf{h}}_k^b||_2^2}{N_k^b} = (s_k^b)_{\text{ UB}},$$

an upper bound for the SINRs among all $b \in \mathcal{B}_k$ is therefore $\max_{b\in\mathcal{B}_k}(s_k^b)_{\text{ UB}}$. Based on (7), an upper bound for the MCS level m for UE k in RB allocation solution $[x_k^b]_{K\times B}$ is:

$$\max\{m | S_m \leq \max_{b\in\mathcal{B}_k} (s_k^b)_{\text{ UB}}, m = 0\cdots M\}. \tag{23}$$

Similarly, to set a lower bound for MCS w.r.t. UE k under a given RB allocation solution $[x_k^b]_{K\times B}$, we can find a lower bound for MCS that can be used on all the RBGs allocated to this UE. Considering the QoS rate requirement Q_k for UE k, the lower bound of the achievable rate of UE k aggregated over all RBGs $b \in \mathcal{B}_k$ cannot be less than Q_k (see Fig. 5(b)). Based on (7), a lower bound for the MCS level m for UE k in RB allocation solution $[x_k^b]_{K\times B}$ is thus:

$$\min\{m | R_m \geq \frac{Q_k}{|\mathcal{B}_k|}, m = 0\cdots M\}. \tag{24}$$

3) Step 3. Feasibility Check: Following Steps 1 and 2, we have a reduced space for $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$. But not every candidate solution $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$ is feasible. Since we are only interested in evaluating the objectives for the feasible solutions, we need to filter out the infeasible solutions first (another step in reducing search space).

To check whether or not a candidate solution pair $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$ is feasible, we check the QoS constraints (16) under ZF beamforming (widely regarded as the optimal beamforming in maximizing throughput). Any candidate solution pair $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$ that cannot meet the QoS constraints (16) will be removed from further considerations (in Step 4).

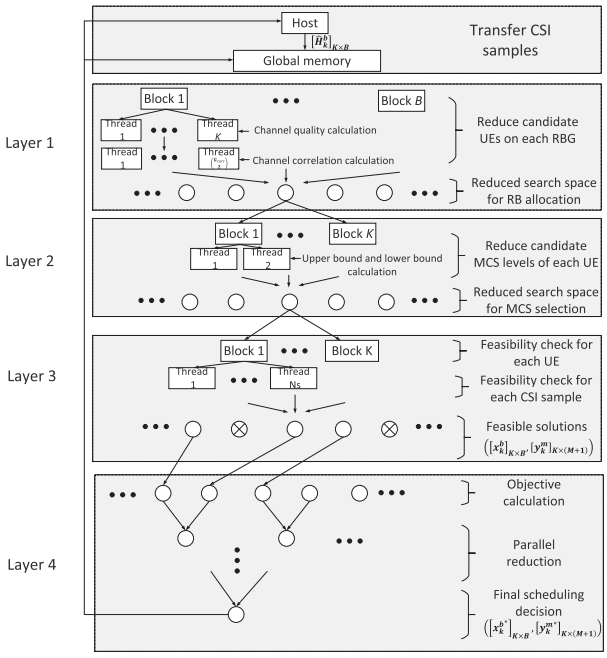


Fig. 6. A parallel implementation of the four steps to solve OPT-D1.

4) Step 4. Finding the Final RB & MCS Solution: Finally, for the remaining feasible decision pairs $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$, we calculate their corresponding objective value in OPT-D1. Then we compare all the achieved objective values by different $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$ pairs and choose the highest objective value for the final solution. The corresponding $([x_k^{b\,*}]_{K\times B}, [y_k^{m*}]_{K\times(M+1)})$ pair will be chosen as the final solution for RB allocation and MCS selection, which will be used as input for the design of the RT control loop in Section IX.

C. Accelerating Computation Time

To ensure the computation time for the above steps can be completed in near-RT (on the order of tens of ms), R^3 employs GPU and leverages its massively parallel processing capability [41]. Figure 6 shows the flow chart of our near-RT implementation on a GPU. At first, the CSI samples $[\hat{H}_k^b]_{K\times B}$ will be transferred from the host to the GPU global memory. In layer 1, B blocks are generated for B independent sub-problems, one for each RBG. For each block, K threads are allocated (one for each UE) to calculate the channel quality based on (21). The UEs with the top G_{rate} highest channel qualities will be chosen as the candidate UEs on each RBG. Then in each block, R^3 generates $\binom{G_{\text{corr}}}{2}$ new threads to calculate the correlation factor between every two UEs based on (22). G_{corr} UE pairs will be chosen, starting with the smallest correlation factors.

In layer 2, for a given RB allocation $[x_k^b]_{K\times B}$, K blocks are generated for K independent sub-problems, one for each UE. For each block, two threads are allocated to calculate the upper bound and lower bound based on (23) and (24), respectively.

In layer 3, for each RB and MCS pair $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$, K blocks are generated for K independent sub-problems, one for each UE. For each block, N_s threads are allocated and each will check one inequality in the LHS of each QoS constraint in (16), one for each CSI sample $\hat{\mathbf{h}}_k^b(j)$.

Finally in layer 4, the objective values in OPT-D1 for all feasible RB and MCS pairs $([x_k^b]_{K\times B}, [y_k^m]_{K\times(M+1)})$ are computed in parallel. Using parallel reduction, R^3 can quickly find the candidate decision pair $([x_k^{b\,*}]_{K\times B}, [y_k^{m*}]_{K\times(M+1)})$ corresponding to the maximum objective value and choose it as the final RB and MCS pair.

IX. A Design of the RT Control Loop

With the RB allocation and MCS selection decision $([(x_k^b)^*]_{K\times B}, [(y_k^m)^*]_{K\times(M+1)})$ found in the near-RT control loop in the last section, R^3 focuses on solving OPT-D2 to find the best beamforming vector $\mathbf{w}_k^b$ in the RT control loop.

A. Basic Idea

To meet the RT constraint (up to 10 ms), R^3 employs the following steps. First, instead of finding a solution for $\mathbf{w}_k^b$ within $\mathbb{C}$, R^3 considers a smaller yet promising search space. Second, for each element in the reduced search space, R^3 performs a feasibility check using the QoS constraints (20). Since it is impossible to check (20) directly due to infinite $\mathbf{c}_k^b(j)$ in $||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b$, we reformulate (20) to an equivalent form for which we only need to check one "worst case" vector (in terms of achievable data rate). Following this step, all infeasible solutions will be dropped from further consideration. Finally, R^3 computes and compares the objective values for all the remaining feasible solutions. The solution $\mathbf{w}_k^{b\,*}$ that yields the highest objective value to OPT-D2 is chosen as the final solution. We elaborate on the details in the following subsections.

B. Search Space

Instead of searching for a beamforming solution $\mathbf{w}_k^b$ in the entire space of $\mathbb{C}$, we focus on a much smaller yet promising search space for $\mathbf{w}_k^b$. Specifically, we only consider beamforming vectors that lie in the spanning space of ZF beamforming vectors. The reason for limiting our focus on the spanning space of ZF beamforming vectors is twofold. First, ZF beamforming vectors, denoted as $(\mathbf{w}_k^b)_j$, can be regarded as an optimal solution for the j-th CSI sample $\hat{\mathbf{h}}_k^b(j)$ w.r.t. throughput [36], as they minimize the overall interference among the UEs. Second, note that in the considered scenario, where each UE is equipped with only one antenna, ZF beamforming won't amplify the noise [36].

For the j-th CSI sample, denote the obtained ZF beamforming matrix as $\mathbf{W}_j = [(\mathbf{w}_k^b)_j]_{K\times B}$ where $(\mathbf{w}_k^b)_j$ is the ZF beamforming vector and is calculated based on $\hat{\mathbf{h}}_k^b(j)$. Then for all N_s CSI samples, a beamforming matrices basis is given as $[\mathbf{W}_1 \cdots \mathbf{W}_{N_s}]^T$. Its spanning space can be generated as $\mathcal{W} = \left\{ a_1\mathbf{W}_1 + \cdots + a_{N_s}\mathbf{W}_{N_s}, a_j \in \mathbb{R}^+, j = 1 \cdots N_s \right\}$. The final reduced search space for $\mathbf{w}_k^b$, denoted as $\mathcal{Z}$ (where each

element $\mathbf{Z} = [\mathbf{z}_k^b]_{K\times B}$), is as follows:

$$\mathcal{Z} = \left\{ \mathbf{Z} \in \mathcal{W} \;\middle|\; \sum_{k=1}^{K} ||\mathbf{z}_k^b||_2^2 = P_{\max}, b = 1 \cdots B \right\} \quad (25)$$

The peak power sum requirement on each RBG b in (25) is intuitive as the maximum throughput is achieved only with maximum transmit power on each RBG b.

C. Feasible Solutions

With the reduced search space $\mathcal{Z}$, the next step is to find all feasible solutions within this space that can satisfy (20). To meet the 10 ms RT requirement, we propose the following procedure.

First, directly checking each inequality in (20) is impractical due to the infinite number of possible $\mathbf{c}_k^b(j)$ that satisfy $||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||2 \le \theta_k^b$. So we transform (20) into an equivalent form (26). In this transformed equation, each inequality requires verification with a specific $\arg\min_{\mathbf{c}_k^b(j)} r_k^b$ on each RBG, enabling a feasible check. Second, determining the data rate $\min_{\mathbf{c}_k^b(j)} r_k^b$ remains challenging in real-time. Instead of finding $\min_{\mathbf{c}_k^b(j)} r_k^b$ directly, we opt to utilize its upper and lower bounds to find the optimal value $\min_{\mathbf{c}_k^b(j)} r_k^b$. Finally, upon computing $\min_{\mathbf{c}_k^b} r_k^b$ on each RBG, we introduce a complete procedure for the feasibility check process. This procedure can be effectively parallelized and executed on a GPU platform, significantly reducing computational time.

1) Reformulating QoS Constraint: Regarding (20), for each candidate beamforming matrix $\mathbf{Z} = [\mathbf{z}_k^b]_{K\times B} \in \mathcal{Z}$, a total of K QoS constraints need to be checked, one for each UE. Within each QoS constraint, N_s inequalities in the LHS of (20) need to be examined, one for each CSI sample. As a result, for each candidate beamforming matrix $\mathbf{Z} = [\mathbf{z}_k^b]_{K\times B} \in \mathcal{Z}$, there are total $N_s K$ inequalities need to be checked.

Based on (20), each inequality–specified by the j-th CSI sample and UE k–should be checked for *every possible* $\mathbf{c}_k^b(j)$ that satisfies $||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b$. This would imply that the inequality be validated for *all* conceivable CSIs within the ambiguity set. This is infeasible—due to the infinite number of possible $\mathbf{c}_k^b(j)$.

Instead of checking (20), we check the following equivalent constraint:

$$\sum_{j=1}^{N_s} \mathbb{I}\left\{ \sum_{b=1}^{B} \min_{\mathbf{c}_k^b(j)} r_k^b \ge Q_k \middle| \mathbf{h}_k^b = \mathbf{c}_k^b(j), ||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b, b = 1 \cdots B \right\} \ge N_s(1-\epsilon) \quad (k = 1 \cdots K). \quad (26)$$

To show (20) and (26) are equivalent, we need to show i) (20) leads to (26) and ii) (26) leads to (20). To show i) is true, we see that if $r_k \ge Q_k$ holds for all $\mathbf{c}_k^b(j)$ s.t. $||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b, b = 1 \cdots B$ in (20), then $r_k \ge Q_k$ also holds for the worst CSI $\arg\min_{\mathbf{c}_k^b(j)} r_k^b$ where $||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b$, $b = 1 \cdots B$. So the corresponding inequality $\sum_{b=1}^{B} \min_{\mathbf{c}_k^b(j)} r_k^b \ge Q_k$ in (26) holds. To show ii) is true, we see that if $\sum_{b=1}^{B} \min_{\mathbf{c}_k^b(j)} r_k^b \ge Q_k$ holds in (26), then $\sum_{b=1}^{B} r_k^b \ge Q_k$ holds since $r_k^b \ge \min_{\mathbf{c}_k^b(j)} r_k^b$ for all $\mathbf{c}_k^b(j)$ s.t. $||\mathbf{c}_k^b(j) - \hat{\mathbf{h}}_k^b(j)||_2 \le \theta_k^b, b = 1 \cdots B$.

There are two benefits in working with (26) instead of (20). First, we only need to consider one specific CSI $\arg\min_{\mathbf{c}_k^b(j)} r_k^b$ for each RBG b, instead of an infinite number of CSI. Second, we can exploit the independence among the RBGs in calculating $\sum_{b=1}^{B} \min_{\mathbf{c}_k^b(j)} r_k^b$. That is, instead of calculating the sum $\sum_{b=1}^{B} \min_{\mathbf{c}_k^b(j)} r_k^b$ at one term, we can calculate each individual term $\min_{\mathbf{c}_k^b(j)} r_k^b$ independently and take the sum afterward.

So the new question becomes: How to calculate $\min_{\mathbf{c}_k^b(j)} r_k^b$ for each RBG b. We now address this question.

2) Calculating $\min_{\mathbf{c}_k^b(j)} r_k^b$*:* To calculate $\min_{\mathbf{c}_k^b(j)} r_k^b$, we still need to carry all the constraints in OPT-D2, except (5). Constraint (5) is automatically satisfied based on (25). We now have the following problem for each RBG b:

$$\textbf{OPT-}r_k^b \quad \min_{\mathbf{c}_k^b(j)} r_k^b \quad \text{s.t. } (6)(7)(8)(19)\ , \quad \mathbf{w}_k^b = \mathbf{z}_k^b, \mathbf{h}_k^b = \mathbf{c}_k^b(j).$$

With a given MCS $(y_k^m)^* = 1$ to UE k (from Section VIII), $\min_{\mathbf{c}_k^b(j)} r_k^b$ in OPT-r_k^b is equivalent to $\min_{\mathbf{c}_k^b(j)} r_k^{b,m^*}$ based on (8), where m^* is the selected MCS level corresponding to $(y_k^m)^* = 1$. Then based on (7), $\min_{\mathbf{c}_k^b(j)} r_k^{b,m}$ is equivalent to $\min_{\mathbf{c}_k^b(j)} s_k^b$. Therefore, solving OPT-r_k^b is equivalent to solving the following problem:

$$\textbf{OPT-}s_k^b \quad \min_{\mathbf{c}_k^b(j)} s_k^b \quad \text{s.t. } (6)(19)\ , \quad \mathbf{w}_k^b = \mathbf{z}_k^b, \mathbf{h}_k^b = \mathbf{c}_k^b(j).$$

But OPT-s_k^b is hard to solve because both (6)(19) are nonlinear.

3) Leveraging Upper and Lower Bounds: Instead of finding the optimal objective value for OPT-s_k^b, denoted as $(s_k^b)_{\text{OPT}}$, we propose to find an upper bound $(s_k^b)_{\text{UB}}$ and a lower bound $(s_k^b)_{\text{LB}}$ for $(s_k^b)_{\text{OPT}}$. Then based on the relative relationships (three cases) between $((s_k^b)_{\text{UB}}, (s_k^b)_{\text{LB}})$ pair and S_{m^*}, we can calculate an upper and lower bound for OPT-r_k^b (denoted as $(r_k^b)_{\text{UB}}$ and $(r_k^b)_{\text{LB}}$) based on (7) and (8), where $(r_k^b)_{\text{UB}}$ and $(r_k^b)_{\text{LB}}$ are either a constant (R_{m^*}) or 0. In each case, we obtain $(r_k^b)_{\text{OPT}}$ easily.

Specifically, referring to Fig. 7, based on the values of $((s_k^b)_{\text{UB}}, (s_k^b)_{\text{LB}})$ pair and their relationship with S_{m^*}, we consider three cases:

- Case 1: If $(s_k^b)_{\text{UB}} < S_{m^*}$, then $(r_k^b)_{\text{UB}} = (r_k^b)_{\text{LB}} = 0$ from (7) and (8). $(r_k^b)_{\text{OPT}}$ can only be 0 since $(r_k^b)_{\text{UB}} = 0$.
- Case 2: If $(s_k^b)_{\text{UB}} \ge S_{m^*}$ and $(s_k^b)_{LB} < S_{m^*}$, then $(r_k^b)_{\text{UB}} = R_{m^*} > 0$ and $(r_k^b)_{\text{LB}} = 0$ from (7) and (8). Then we can set $(r_k^b)_{\text{OPT}}$ to 0 to be conservative since $(r_k^b)_{\text{LB}} = 0$.
- Case 3: If $(s_k^b)_{\text{LB}} \ge S_{m^*}$, then $(r_k^b)_{\text{LB}} = (r_k^b)_{\text{UB}} = R_{m^*} > 0$ from (7) and (8), $(r_k^b)_{\text{OPT}}$ must be R_{m^*}.

Based on the above discussions, we now only need to find $(s_k^b)_{\text{UB}}$ and $(s_k^b)_{\text{LB}}$.

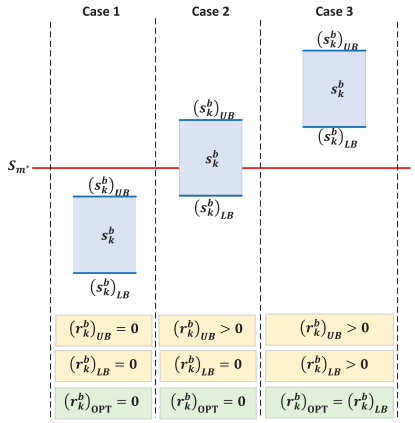


Fig. 7. Three cases encountered in finding the optimal objective value $(r_k^b)_{\text{OPT}}$.

4) Finding Upper Bound $(s_k^b)_{UB}$: Since OPT-s_k^b is a minimization problem, then any *feasible* solution to OPT-s_k^b can serve as a UB solution to $\min_{\mathbf{c}_k^b(j)} s_k^b$. To find a feasible solution, we can solve

$$\min_{\mathbf{c}_k^b(j)} |(\mathbf{z}_k^b)^\dagger \mathbf{c}_k^b(j)|^2 \quad \text{s.t. (19)}, \tag{27}$$

where the objective function in (27) is the numerator in the RHS of (6). A feasible solution to (27) can be used as a UB solution to OPT-s_k^b.

Given that (27) is a convex optimization, we can easily find an optimal solution $\mathbf{c}_k^b(j)^*$ as follows:

$$\mathbf{c}_k^b(j)^* = \hat{\mathbf{h}}_k^b(j) - \theta_k^b \frac{\mathbf{z}_k^b}{||\mathbf{z}_k^b||_2} e^{i\cdot \arg\left((\mathbf{z}_k^b)^\dagger \hat{\mathbf{h}}_k^b(j)\right)}, \tag{28}$$

where $\arg(\cdot)$ is the phase function and i is the imaginary unit. With $\mathbf{c}_k^b(j)^*$, an upper bound $(s_k^b)_{\text{UB}}$ can be obtained by plugging $\mathbf{c}_k^b(j)^*$ into (6).

5) Finding Lower Bound $(s_k^b)_{LB}$: To find a LB for $\min_{\mathbf{c}_k^b(j)} s_k^b$, we have:

$$\begin{aligned} \min_{\mathbf{c}_k^b(j)} \; s_k^b &\geq \frac{\min_{\mathbf{c}_k^b(j)} |(\mathbf{z}_k^b)^\dagger \mathbf{c}_k^b(j)|^2}{\max_{\mathbf{c}_k^b(j)} \sum_{i=1}^{K, i\neq k} |(\mathbf{z}_i^b)^\dagger \mathbf{c}_k^b(j)|^2 + N_k^b} \\ &\geq \frac{\min_{\mathbf{c}_k^b(j)} |(\mathbf{z}_k^b)^\dagger \mathbf{c}_k^b(j)|^2}{\sum_{i=1}^{K, i\neq k} \max_{\mathbf{c}_k^b(j)} |(\mathbf{z}_i^b)^\dagger \mathbf{c}_k^b(j)|^2 + N_k^b}, \end{aligned} \tag{29}$$

where the first inequality comes from (6). In the denominator of (29), we can decompose $\sum_{i=1}^{K, i\neq k} \max_{\mathbf{c}_k^b(j)} |(\mathbf{z}_i^b)^\dagger \mathbf{c}_k^b(j)|^2$ into $(K-1)$ independent maximization problems, with i-th problem in the following form:

$$\max_{\mathbf{c}_k^b(j)} |(\mathbf{z}_i^b)^\dagger \mathbf{c}_k^b(j)|^2 \quad \text{s.t. (19)}. \tag{30}$$

Given that (30) is a convex optimization, we can easily find an optimal solution $\mathbf{c}_k^b(j)_i^*$ as follows:

$$\mathbf{c}_k^b(j)_i^* = \hat{\mathbf{h}}_k^b(j) + \theta_k^b \frac{\mathbf{z}_i^b}{||\mathbf{z}_i^b||_2} e^{i\cdot \arg\left((\mathbf{z}_i^b)^\dagger \hat{\mathbf{h}}_k^b(j)\right)} \tag{31}$$

Combining (28) for the numerator in (29), we have a LB for $\min_{\mathbf{c}_k^b(j)} s_k^b$ as follows:

$$(s_k^b)_{\text{LB}} = \frac{|(\mathbf{z}_k^b)^\dagger \mathbf{c}_k^b(j)^*|^2}{\sum_{i=1}^{K, i\neq k} |(\mathbf{z}_i^b)^\dagger \mathbf{c}_k^b(j)_i^*|^2 + N_k^b}. \tag{32}$$

Once we have $(s_k^b)_{\text{UB}}$ and $(s_k^b)_{\text{LB}}$, we can consider one of the three cases in Fig. 7 and find $(r_k^b)_{\text{OPT}}$ for OPT-r_k^b.

Algorithm 1 Feasibility Check of $\mathbf{Z} \in \mathcal{Z}$

Input: $\mathcal{Z}$, $[\hat{\mathbf{H}}_k^b]_{K\times B}$, $[\theta_k^b]_{K\times B}$, $[(x_k^b)^*]_{K\times B}$, $[(y_k^m)^*]_{K\times(M+1)}$.
Output: All feasible $\mathbf{Z} \in \mathcal{Z}$ that satisfy (26)

1 **foreach** $\mathbf{Z} \in \mathcal{Z}$ **do**
2 **foreach** $k = 1 \cdots K$ **do**
3 $I_k = 0$;
4 **foreach** $j = 1 \cdots N_s$ **do**
5 $r_k = 0$;
6 **foreach** $b = 1 \cdots B$ **do**
7 Calculate $(s_k^b)_{\text{UB}}$ based on (28);
8 Calculate $(s_k^b)_{\text{LB}}$ based on (32);
9 **if** $(s_k^b)_{UB} < S_{m^*}$ **then**
10 $(r_k^b)_{\text{OPT}} = 0$;
11 **else if** $(s_k^b)_{UB} \geq S_{m^*}$ *and* $(s_k^b)_{LB} < S_{m^*}$ **then**
12 $(r_k^b)_{\text{OPT}} = 0$;
13 **else**
14 $(r_k^b)_{\text{OPT}} = R_{m^*}$;
15 $r_k := r_k + (r_k^b)_{\text{OPT}}$;
16 **if** $r_k \geq Q_k$ **then**
17 $I_k := I_k + 1$;
18 **if** $I_k \geq N_s(1-\epsilon)$ **then**
19 (26) is satisfied for UE k;
20 **if** *(26) is satisfied for all UEs* $k = 1 \cdots K$ **then**
21 $\mathbf{Z}$ is feasible;
22 **else**
23 $\mathbf{Z}$ is infeasible;

6) A Recap of Complete Procedure: Algorithm 1 summarizes the complete procedure of finding all feasible beamforming matrix $\mathbf{Z} \in \mathcal{Z}$ that satisfy QoS constraints (26). In Algorithm 1, there are four nested loops, organized from the outermost to the innermost as follows:

Outermost Loop (First Loop): The feasibility of each candidate beamforming matrix, denoted as $\mathbf{Z} \in \mathcal{Z}$, is evaluated w.r.t. the QoS constraints (26), encompassing a total of $|\mathcal{Z}|$ candidates (Lines 1–33).

Second Loop: For a given candidate beamforming matrix $\mathbf{Z} \in \mathcal{Z}$, each of the K QoS constraints specified in (26) is evaluated (Lines 2–19).

Third Loop: For each given candidate beamforming matrix $\mathbf{Z} \in \mathcal{Z}$ and UE k, each inequality on the LHS of (26), amounting to a total of N_s inequalities, is verified (Lines 4–17).

Innermost Loop (Fourth Loop): For a given candidate beamforming matrix $\mathbf{Z} \in \mathcal{Z}$, UE k and j-th CSI sample, each term $\min_{\mathbf{c}_k^b(j)} r_k^b$ on the LHS of every inequality in (26), totaling B terms, is computed (Lines 6–15). Specifically, $(s_k^b)_{\text{UB}}$ for OPT-s_k^b is calculated based on (28) (Line 7), $(s_k^b)_{\text{LB}}$ for OPT-s_k^b is calculated based on (32) (Line 8). Then based on the three cases in Fig. 7, $(r_k^b)_{\text{OPT}}$ for $\min_{\mathbf{c}_k^b(j)} r_k^b$ can be obtained (Lines 9-14).

Algorithm 1 can be efficiently parallelized due to the inherent independence among the $|\mathcal{Z}|$ candidate beamforming matrix, K UEs, N_s CSI samples, and B RBGs. This means that a total of $|\mathcal{Z}| \cdot K \cdot N_s \cdot B$ evaluations can be done in parallel.

D. Finding Final Solution

Finally, for all the feasible solutions $\mathbf{Z} \in \mathcal{Z}$, R^3 will calculate their corresponding objective values in OPT-D2 and the one with the highest objective value will be chosen as the final scheduling decision for beamforming.

E. Accelerating Computation Time

To ensure the computation time can be completed in RT (less than 10 ms), R^3 again employs GPU for acceleration. Figure 8 shows the flow chart of our RT implementation on a GPU. At first, the CSI samples $[\hat{H}_k^b]_{K\times B}$ and RB allocation/MCS selection decision $([x_k^{b*}]_{K\times B}, [y_k^{m*}]_{K\times(M+1)})$ from OPT-D1 in the near-RT control loop will be transferred from the host to the GPU global memory. In layer 1, Ns blocks are generated for Ns independent sub-problems, one for each CSI sample. For each block, one thread is used to calculate the ZF beamforming matrix based on each CSI sample. Then all N_s ZF beamforming matrices are used to generate the reduced search space based on (25).

In layer 2, a total of $|\mathcal{Z}| \cdot K \cdot N_s \cdot B$ threads are generated for the feasibility check process shown in Algorithm 1.

Finally, in layer 3, the objective values in OPT-D2 for all feasible beamforming matrices $\mathbf{Z}$ are computed in parallel. Using parallel reduction, R^3 can quickly find the best decision matrix $\mathbf{Z}^*$ corresponding to the maximum objective value and choose it as the final beamforming solution.

X. Experimental Results

In this section, we present a comprehensive evaluation of R^3 through a series of simulation experiments. We implement R^3 on NVIDIA RTX 4090 GPU [45] with CUDA 12.0 Toolkit [46], [47]. The Nvidia RTX 4090 is a consumer-grade GPU with 16,384 CUDA cores and an operating frequency of 2.23 GHz.

We will assess the performance of R^3 through three key metrics: violation rate of QoS constraint, throughput (objective value), and computation time (for the RT control loop). We first use a case study to demonstrate R^3's behavior and performance. Then we investigate how R^3 behaves under various parameter settings.

As for comparison, we find that none of the existing works address exactly the same problem as ours in this paper. Therefore, we have to make the necessary customization for the state-of-the-art to make a meaningful comparison. Specifically, we will compare R^3 with the following two alternative designs.

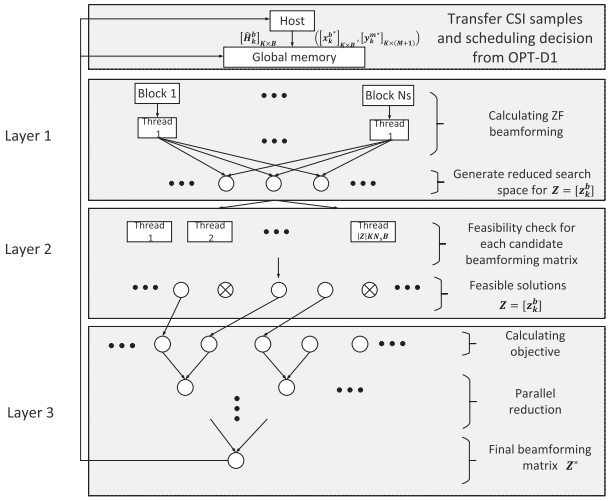


Fig. 8. A parallel implementation to solve OPT-D2.

- **Unified Scheduling** [21]. Unified scheduling addresses RB allocation and MCS selection different from our R^3. It employs an iterative greedy algorithm, aiming to maximize throughput in each iteration. Given that Unified scheduling assumes knowledge of perfect CSI, which is not available in our problem, we will use the mean CSI sample as an approximation. Furthermore, as Unified scheduling doesn't address beamforming, we employ ZF for a fair comparison.
- **Gaussian Approximation** [28]. Gaussian approximation is another (different) technique that can be used to address chance constraints. It transforms a chance constraint into a deterministic constraint while maintaining a probabilistic guarantee. This method presumes that the channel adheres to a Gaussian distribution, with both its mean and variance derived from the CSI samples. Since Gaussian approximation pertains solely to the chance constraint, we assume all other components (RB allocation, MCS selection, and beamforming) will align with R^3 so as to make a fair comparison.

It is important to note that both the Unified Scheduling and Gaussian Approximation algorithms are implemented directly on the GPU, with iterations executed sequentially. Optimizing these implementations to maximize parallelism is beyond the scope of this paper.

A. Simulation Settings

The set of common parameters that will be used in all of our simulation experiments is given in Table II. Parameters not listed in the table will be specified in a given study.

The UEs are randomly located within the cell radius (see Fig. 9), with the number of UEs to be specified in each study. The channel is modeled using a combination of path loss and fast fading. The path loss is calculated based on $38+30\log_{10}(d)$ [48], where d represents the distance between the UE and the O-RU. We generate the fast-fading component

TABLE II
PARAMETER SETTINGS

Parameter	Value
Numerology	0
Number of RBs (B)	100
Number of RBGs	25 (4 RBs in each RBG)
Bandwidth of one RB (BW)	180 KHz
Maximum MCS level (M)	28
Number of transmit antennas (A_T)	8
Power budget ($P_{\max}$)	33 dbm
CSI time window (L)	100
QoS data requirement (Q_k)	5–10 Mbps
Candidate UEs based on data rate (G_{rate})	10
Candidate UEs based on correlation (G_{corr})	4
Maximum noise power (θ_k^b)	-150 dbm
Cell radius	300 m

of the channel using the Rayleigh distribution [43]. It is important to clarify that the distributions described above are solely utilized to generate the channel behavior. This information is not available and is not utilized by our R^3 algorithm, which solely relies on the limited CSI samples (with zero knowledge or any assumption of the underlying distribution information).

The time length of each CSI sample is set to 1 TTI (1 ms). The CSI time window L is set to 100 ms. We assume the channel distribution remains stable within this window. This assumption can be justified by the actual channel behavior (in terms of path loss and fast fading) in the real world as follows.

- For path loss, let's consider a scenario with a rapidly moving vehicle traveling at 30 m/s (65 miles/hour) and located 50 m away from the O-RU (d=50 m). Based on the aforementioned path loss model, the variation in path loss is less than 1 dB (from 88.9 dB to 89.7 dB) over a 100 ms window. This suggests that the path loss remains relatively constant within this window.
- For fast fading, its distribution is predominantly determined by the Doppler shift, which can be represented by $\frac{vf}{c}$, where v denotes the UE's velocity, f is the carrier frequency, and c is the speed of light [44]. As an example, consider a UE with a carrier frequency $f = 2$ GHz. Suppose it is traveling at an initial speed of 20 m/s (45 miles/hour) and is accelerating at 6 m/s^2. Then over a 100 ms period, the Doppler shift varies from 133 Hz to 137 Hz, a mere change of only 4 Hz. This indicates that the change in the fast fading distribution over a 100 ms duration is negligible.

In summary, given the negligible changes in both path loss and fast fading components within a 100 ms period, we can confidently conclude that the channel distribution remains fairly consistent (relatively unchanged) for the $L = 100$ ms window. Consequently, all the CSI samples collected within this period can be assumed to share the same channel distribution.

B. A Case Study

In this study, we set the number of UEs $K = 20$ as shown in Fig. 9. We set $N_s = 50$, which gives us $\gamma = 0.5$. We run our simulation for 1,000 seconds and take an average of the results over all scheduling instances. In the simulation results, confidence interval is not included. This is because, during the 1,000-second simulation period, the channel distribution is both unknown and time-varying while confidence interval is only meaningful for stationary distributions.

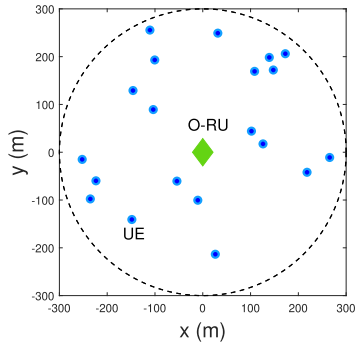


Fig. 9. Network topology used for the case study.

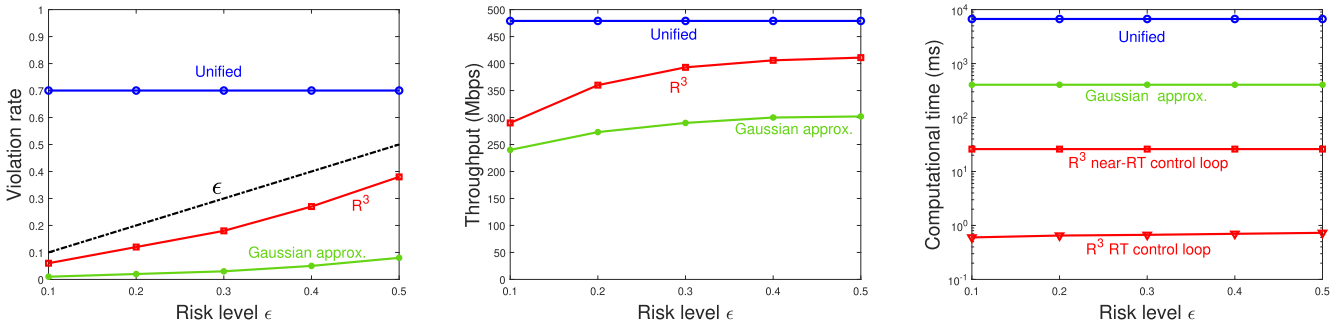
1) Statistical QoS Guarantee: In Fig. 10(a), we present the actual violation rates of the QoS constraints for all three algorithms. The violation rate is computed by taking the average across the violation rates of QoS constraints for all UEs. The violation rates of R^3 and Gaussian Approximation remain consistently below the target risk level ϵ. This indicates that these algorithms can satisfy the required QoS constraints. Notably, the violation rates of both R^3 and Gaussian approximation increase as the risk level increases and Gaussian approximation exhibits a more conservative behavior than R^3. In contrast, Unified scheduling has a fixed prohibitively high violation rate and cannot meet the required QoS constraints. This is because it intends to allocate more RBGs and high MCS levels to UEs with high channel quality for higher throughput without considering the QoS requirements.

2) Throughput: Figure 10(b) shows the system throughput (sum over all UEs) of all three algorithms. Unified scheduling has the highest throughput since it tends to allocate most of the resources to UEs which have better channel quality. R^3 demonstrates the second-highest performance, with only a 15% gap compared to unified scheduling when $\epsilon \geq 0.3$. Gaussian approximation has the lowest throughput. Moreover, the throughput of both R^3 and Gaussian approximation increases as the risk level rises due to the larger feasible region.

3) Computation Time: Figure 10(c) shows the computation time of all three algorithms. Unified scheduling has the longest computation time $\sim$10 seconds because it follows an iterative greedy method. The computation time of Gaussian approximation is $\sim$300 ms. The near-RT control loop of R^3 takes $\sim$ 30ms and meets the time requirement for the near-RT control loop. The computation time of the RT control loop of R^3 is less than 1 ms, which is well within the time requirement for RT control loop (less than 10 ms). The computational time of R^3 escalates as the risk level rises, because of an expanded feasible search space for making scheduling decisions.

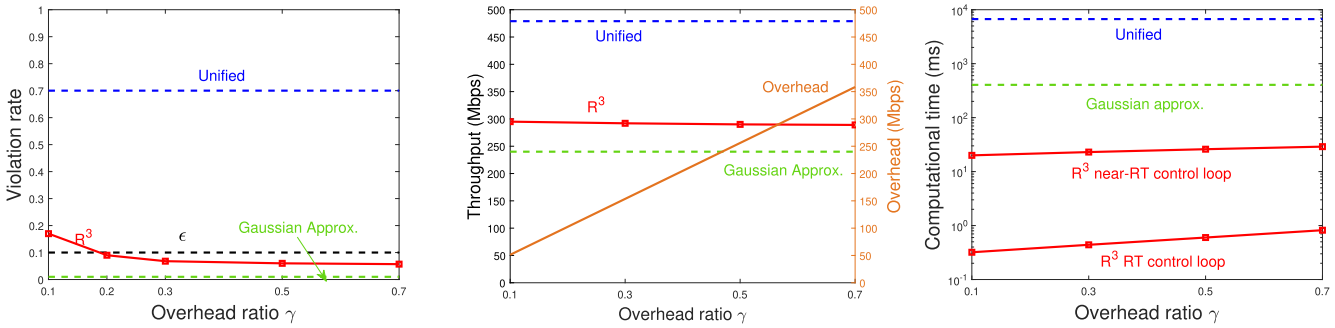
C. Varying System Parameters

1) Varying Overhead Ratio γ: A larger value of γ means more CSI samples (N_s) can be utilized. Given that γ is the distinct parameter for R^3, the performance of Unified



(a) QoS guarantee (b) Throughput (c) Computation time

Fig. 10. Performance of R^3 in a case study.



(a) QoS guarantee (b) Throughput and Overhead (c) Computation time

Fig. 11. Performance of R^3 under varying value of γ.

scheduling and Gaussian approximation from the case study is not affected by varying γ. All other simulation parameters are the same as those in the case study, with the exception that the risk level, ϵ, is set to a fixed value of 0.1.

In Fig. 11(a), we present the actual violation rates of the QoS constraints under varying γ. When γ increases, which means more CSI samples can be utilized, the actual risk level decreases because the empirical distribution approaches the true distribution, and the performance of R^3 improves. When $\gamma \leq 0.2$, the actual violation rate is higher than the risk level ϵ, which means the number of CSI samples is insufficient to achieve the desired probabilistic guarantee for R^3. When $\gamma = 0.2$, the violation rate approximately equals to the target risk level. When $\gamma \geq 0.2$, the violation rates of R^3 remain consistently lower than the target risk level.

Figure 11(b) shows throughput (left vertical axis) and overhead (right vertical axis) as a function of varying γ. The overhead can be calculated as # of UEs $\times$ # of RB(G)s $\times$ # of antennas $\times$ float type (e.g., 32 bits) $\times$ γ. With a larger γ, the actual overhead in the control channel used for transmitting CSI samples through the fronthaul link increases. On the other hand, throughput under different values of γ remains almost the same. This is because the data channel and control channel are separate and the throughput is rather immune to the value of γ.

Figure 11(c) shows the computation time of R^3 under varying γ. When γ increases, which means more CSI samples are being utilized, the computational time increases. But for $\gamma \leq 0.7$, the computational time of R^3 meets the timing requirement of O-RAN in both the near-RT and RT control loops.

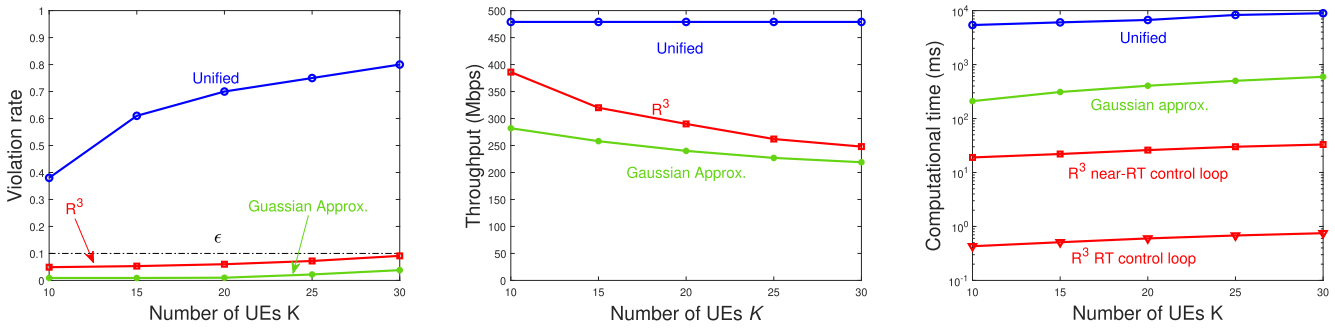
2) Varying Number of UEs K: A larger K means more QoS constraints need to be satisfied when making scheduling decisions. In this study, all other simulation parameters are the same as those in the case study, with the exception that $\epsilon = 0.1$.

In Fig. 12(a), we present the actual violation rates of the QoS constraints under varying K. When K increases, the actual violation rates of all three algorithms increase. The violation rates of R^3 and Gaussian approximation remain consistently lower than the risk level $\epsilon = 0.1$. The violation rates of Unified scheduling significantly exceed the target risk level $\epsilon = 0.1$.

Figure 12(b) shows the throughput (objective value) as a function of K. With an increasing value of K, the throughput of R^3 and Gaussian approximation decreases. This is because as the number of UEs increases, the number of QoS constraints also increases, leading to a reduced feasible region and consequently a lower objective value. The throughput of Unified scheduling remains static because it allocates resources only to a subset of UEs exhibiting good channel quality.

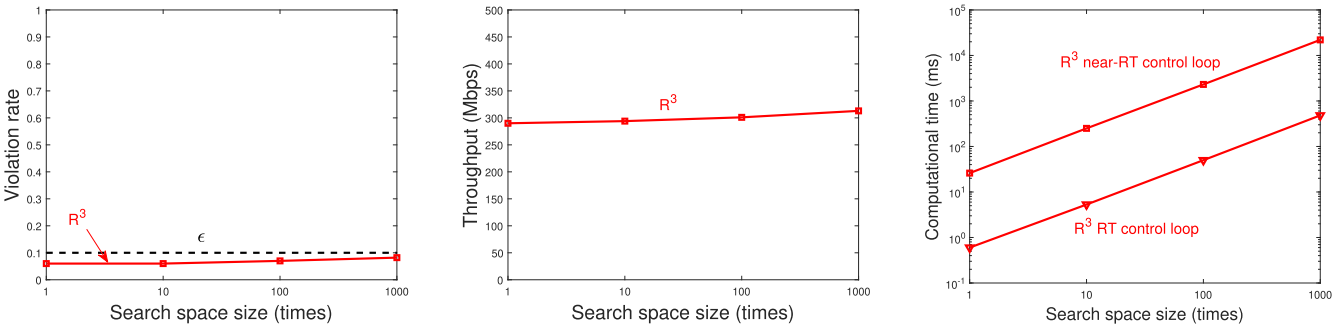
Figure 12(c) shows the computation time of all three algorithms as a function of K. The computational time of all three algorithms increases as the number of UE increases. The computation time of R^3 (near-RT control loop and RT control loop) both satisfy their timing requirements in O-RAN.

3) Varying Size of Search Space: In the case study, after the search space reduction process in R^3, the search space size of



(a) QoS guarantee (b) Throughput (c) Computation time

Fig. 12. Performance of R^3 under varying number of UEs K.



(a) QoS guarantee (b) Throughput (c) Computation time

Fig. 13. Performance of R^3 under varying search space size.

both the near-real-time control loop and the real-time control loop is reduced to the order of 10^4. To evaluate the impact of the search space reduction process, we use the reduced search space used in the case study (which meets our real-time requirement) as the baseline. Then we consider larger search spaces with $10\times$, $100\times$, and $1000\times$ of the baseline search space. Note that the original search space cannot be evaluated due to its prohibitively large size, which can include up to 10^{120} possible solutions. All other simulation parameters are the same as those in the case study, with the exception that the risk level ϵ is set to 0.1.

In Fig. 13(a), we present the actual violation rates of the QoS constraints as a function of the search space size. For different-sized search spaces, the actual violation rates remain below the risk level. As the search space size increases, the violation rates become slightly closer to the risk level.

Figure 13(b) shows the throughput (objective value) as a function of the size of search space. Larger search spaces can achieve higher throughput, which is intuitive. However, when the search space is $1{,}000\times$ of the baseline, the throughput improvement is only 8%, indicating that our baseline search space is excellent (in terms of containing high-quality solutions).

Figure 13(c) shows the computation time as a function of the search space size. The computation time increases nearly linearly (log scale for both x and y axes) with the search space size. Only the baseline reduced search space of the RT control loop can meet the sub-1 ms real-time requirement. When the search space is $1{,}000\times$ of the baseline, the computation time of the RT control loop is $\sim$600 ms, far exceeding the 1 ms RT requirement.

As the results show, R^3 achieves an excellent trade-off between actual performance and computational time, indicating that the reduced search space is of high quality.

XI. Conclusion

We presented R^3—a real-time robust MU-MIMO scheduler. To date, R^3 is the only scheduler in the field that offers a comprehensive solution for RB(G) allocation, MCS assignment, and beamforming calculation within the O-RAN framework. In particular, R^3 is the only known scheduler that successfully addresses the two prominent challenges to scheduler design in O-RAN: imperfect CSI and real-time requirement. To address the imperfect CSI, R^3 uses a data-driven approach and utilizes the limited number of CSI samples to provide probabilistic guarantees. To meet the stringent time requirement of the O-RAN system, R^3 decomposes the scheduling problem into two distinct sub-problems and fits them into near-RT and RT control loops of O-RAN. Each sub-problem is designed with a parallel structure with reduced search space and implemented on the GPU platform. Experiment results confirm that R^3 meets our design objectives over a wide range of operating conditions.

References

[1] (2023). *O-RAN Alliance*. Accessed: Dec. 20, 2023. [Online]. Available: https://www.o-ran.org

[2] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 1376–1411, 2nd Quart., 2023.

[3] A. Garcia-Saavedra and X. Costa-Perez, "O-RAN: Disrupting the virtualized RAN ecosystem," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 96–103, Oct. 2021.

[4] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and learning in O-RAN for data-driven NextG cellular networks," *IEEE Commun. Mag.*, vol. 59, no. 10, pp. 21–27, Oct. 2021.

[5] A. S. Abdalla, P. S. Upadhyaya, V. K. Shah, and V. Marojevic, "Toward next generation open radio access networks: What O-RAN can and cannot do!" *IEEE Netw.*, vol. 36, no. 6, pp. 206–213, Nov. 2022.

[6] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, "The road towards 6G: A comprehensive survey," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 334–366, 2021.

[7] O-RAN Alliance. (Apr. 2023). *Overview of Open Testing and Integration Centre (OTIC) and O-RAN Certification and Badging Program*. Accessed: Dec. 20, 2023. [Online]. Available: https://www.o-ran.org/resources

[8] O-RAN Alliance. (Jun. 2023). *O-RAN Working Group 3 (Near-Real-Time RAN Intelligent Controller and E2 Interface WG) Near-RT RIC Architecture (O-RAN.WG3.RICARCH-R003-v05.00)*. Accessed: Dec. 20, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[9] O-RAN Alliance. (Jun. 2023). *O-RAN Working Group 2 (Non-RT RIC and A1 Interface WG) Non-RT RIC Architecture (O-RAN.WG2.Non-RT-RIC-ARCH-R003-v04.00)*. Accessed: Dec. 20, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[10] O-RAN Alliance. (Jun. 2023). *O-RAN Working Group 1 (Use Cases and Overall Architecture) O-RAN Architecture Description (O-RAN.WG1.OAD-R003-v09.00)*. Accessed: Dec. 20, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[11] Y. Xu, X. Zhao, and Y.-C. Liang, "Robust power control and beamforming in cognitive radio networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1834–1857, 4th Quart., 2015.

[12] A. F. Molisch et al., "Hybrid beamforming for massive MIMO: A survey," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 134–141, Sep. 2017.

[13] I. Cale, A. Salihovic, and M. Ivekovic, "Gigabit passive optical network–GPON," in *Proc. 29th Int. Conf. Inf. Technol. Interface*, Jun. 2007, pp. 679–684.

[14] O-RAN Alliance. *Transport Layer and O-RAN Fronthaul Protocol Implementation*. Accessed: Dec. 20, 2023. [Online]. Available: https://docs.o-ran-sc.org/projects/o-ran-sc-o-du-phy/en/latest

[15] M. Peng, Y. Sun, X. Li, Z. Mao, and C. Wang, "Recent advances in cloud radio access networks: System architectures, key techniques, and open issues," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2282–2308, 3rd Quart., 2016.

[16] O-RAN Alliance. (Jun. 2023). *O-RAN Working Group 4 (Open Fronthaul Interfaces WG) Control, UE and Synchronization Plane Specification (O-RAN.WG4.CUS.0-R003-v12.00)*. Accessed: Dec. 20, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[17] Y. Chen, Y. Wu, Y. T. Hou, and W. Lou, "MCore: Achieving sub-millisecond scheduling for 5G MU-MIMO systems," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Vancouver, BC, Canada, May 2021, pp. 1–10.

[18] Y. Huang, S. Li, Y. T. Hou, and W. Lou, "GPF: A GPU-based design to achieve $\sim$100 μs scheduling for 5G NR," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, New Delhi, India, Oct. 2018, pp. 207–222.

[19] Y. Chen, Y. T. Hou, W. Lou, J. H. Reed, and S. Kompella, "M ^{3}: A sub-millisecond scheduler for multi-cell MIMO networks under C-RAN architecture," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, London, U.K., May 2022, pp. 130–139.

[20] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4331–4340, Sep. 2011.

[21] H. Zhang, N. Prasad, and S. Rangarajan, "MIMO downlink scheduling in LTE systems," in *Proc. IEEE INFOCOM*, Orlando, FL, USA, Mar. 2012, pp. 2936–2940.

[22] S.-B. Lee, I. Pefkianakis, S. Choudhury, S. Xu, and S. Lu, "Exploiting spatial, frequency, and multiuser diversity in 3GPP LTE cellular networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 11, pp. 1652–1665, Nov. 2012.

[23] A. Ragaleux, S. Baey, and A. Fladenmuller, "An efficient and generic downlink resource allocation procedure for pre-5G networks," *Wireless Commun. Mobile Comput.*, vol. 16, no. 17, pp. 3089–3103, Dec. 2016.

[24] Y. Xu, H. Yang, F. Ren, C. Lin, and X. S. Shen, "Frequency domain packet scheduling with MIMO for 3GPP LTE downlink," *IEEE Trans. Wireless Commun.*, vol. 12, no. 4, pp. 1752–1761, Apr. 2013.

[25] J. Wang, S. Jin, X. Gao, K.-K. Wong, and E. Au, "Statistical eigenmode-based SDMA for two-user downlink," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5371–5383, Oct. 2012.

[26] V. Raghavan, S. V. Hanly, and V. V. Veeravalli, "Statistical beamforming on the Grassmann manifold for the two-user broadcast channel," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6464–6489, Oct. 2013.

[27] S. Li et al., "Maximize spectrum efficiency in underlay coexistence with channel uncertainty," *IEEE/ACM Trans. Netw.*, vol. 29, no. 2, pp. 764–778, Apr. 2021.

[28] K. Wang, A. M. So, T. Chang, W. Ma, and C. Chi, "Outage constrained robust transmit optimization for multiuser MISO downlinks: Tractable approximations by conic optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 21, pp. 5690–5705, Nov. 2014.

[29] M. B. Shenouda, T. N. Davidson, and L. Lampe, "Outage-based design of robust Tomlinson Harashima transceivers for the MISO downlink with QoS requirements," *IEEE Trans. Signal Process.*, vol. 93, no. 12, pp. 3341–3352, Dec. 2013.

[30] Y. Shi, J. Zhang, and K. B. Letaief, "Optimal stochastic coordinated beamforming for wireless cooperative networks with CSI uncertainty," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 960–973, Feb. 2014.

[31] S. Acharya et al., "Mitra: An O-RAN based real-time solution for coexistence between general and priority users in CBRS," in *Proc. IEEE 20th Int. Conf. Mobile Ad Hoc Smart Syst. (MASS)*, Toronto, ON, Canada, Sep. 2023, pp. 295–303.

[32] S. Li, N. Jiang, Y. Chen, Y. T. Hou, W. Lou, and W. Xie, "D2BF—Data-driven beamforming in MU-MIMO with channel estimation uncertainty," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, London, U.K., May 2022, pp. 120–129.

[33] 3GPP. (Jun. 2023). *NR; Physical Layer Procedures for Data (TS 38.214 Version 17.6.0)*. Accessed: Dec. 20, 2023. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications

[34] J. R. Birge and F. V. Louveaux, *Introduction to Stochastic Programming*. New York, NY, USA: Springer, 1997, ch. 2.

[35] Y. Huang, S. Li, Y. T. Hou, and W. Lou, "GPF+: A novel ultrafast GPU-based proportional fair scheduler for 5G NR," *IEEE/ACM Trans. Netw.*, vol. 30, no. 2, pp. 601–615, Apr. 2022.

[36] N. Jindal, "MIMO broadcast channels with finite-rate feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045–5060, Nov. 2006.

[37] S. S. Christensen, R. Agarwal, E. De Carvalho, and J. M. Cioffi, "Weighted sum-rate maximization using weighted MMSE for MIMO-BC beamforming design," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4792–4799, Dec. 2008.

[38] D. Bertsimas, V. Gupta, and N. Kallus, "Robust sample average approximation," *Math. Program.*, vol. 171, pp. 217–282, Sep. 2018.

[39] T. Champion, L. De Pascale, and P. Juutinen, "The ∞-Wasserstein distance: Local solutions and existence of optimal transport maps," *SIAM J. Math. Anal.*, vol. 40, no. 1, pp. 1–20, Jan. 2008.

[40] W. Xie, "On distributionally robust chance constrained programs with Wasserstein distance," *Math. Program.*, vol. 186, no. 1, pp. 115–155, Mar. 2021.

[41] Y. Huang et al., "GPU: A new enabling platform for real-time optimization in wireless networks," *IEEE Netw.*, vol. 34, no. 6, pp. 77–83, Jun. 2020.

[42] 3GPP. (Jun. 2021). *NR; Physical Channels and Modulation (TR 36.211 Version 16.6.0)*. Accessed: Dec. 20, 2023. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications

[43] A. M. Tulino, G. Caire, S. Shamai, and S. Verdu, "Capacity of channels with frequency-selective and time-selective fading," *IEEE Trans. Inf. Theory*, vol. 56, no. 3, pp. 1187–1215, Mar. 2010.

[44] 3GPP. (2023). *Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) Radio Transmission and Reception (TS 36.101 Version 18.3.0)*. Accessed: Dec. 20, 2023. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications

[45] NVIDIA. *RTX 4090*. Accessed: Dec. 20, 2023. [Online]. Available: https://www.nvidia.com/en-us/geforce/graphics-cards/40-series/rtx-4090/

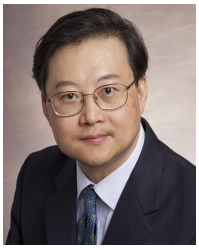
[46] NVIDIA. (2023). *CUDA C Programming Guide V10.2.89*. Accessed: Dec. 20, 2023. [Online]. Available: https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html

[47] NVIDIA. *CUDA Toolkit*. Accessed: Dec. 20, 2023. [Online]. Available: https://developer.nvidia.com/CUDA-toolkit

[48] 3GPP. (Mar. 2022). *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Frequency (RF) Requirements for LTE Pico Node B (TS 36.931 Version 17.7.0)*. Accessed: Dec. 20, 2023. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications

Yubo Wu (Student Member, IEEE) received the B.E. and M.S. degrees in telecommunication engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2016 and 2019, respectively. He is currently pursuing the Ph.D. degree with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA. His current research interests include automotive radar signal processing for autonomous vehicles, wireless networking, and novel GPU-based designs and implementation to accelerate computation for real-time applications.

Yi Shi (Senior Member, IEEE) received the Ph.D. degree in computer engineering from Virginia Tech, Blacksburg, VA, USA, in 2007. He is currently a Research Associate Professor with Commonwealth Cyber Initiative (CCI) and a Research Associate Professor of electrical and computer engineering (by courtesy) at Virginia Tech, Arlington, VA, USA. Before joining Virginia Tech in 2022, he was a Senior Lead Scientist at Intelligent Automation, a BlueHalo company, Rockville, MD, USA. His papers have appeared in top-tier IEEE/ACM journals and international conferences. His research interests include algorithm designs, optimization, and machine learning for next-G wireless networks. He was a recipient of the IEEE INFOCOM Test of Time Paper Award in 2023, the Best Paper Award at IEEE HST 2018, the Best Student Paper Award at ACM WUWNet 2014, the only Best Paper Award Runner-Up at IEEE INFOCOM 2011, and the Best Paper Award at IEEE INFOCOM 2008. He is an Editor of IEEE COMMUNICATIONS SURVEYS AND TUTORIALS and IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING. He also served as the TPC Chair for a number of IEEE and ACM symposiums, tracks, and workshops.



Y. Thomas Hou (Fellow, IEEE) received the Ph.D. degree from NYU Tandon School of Engineering in 1998. He is currently Bradley Distinguished Professor of electrical and computer engineering at Virginia Tech, Blacksburg, VA, USA. He has published over 350 papers in IEEE/ACM journals and conferences. His current research interests include developing real-time optimal solutions to complex science and engineering problems arising from wireless and mobile networks. He is also interested in wireless security. His papers were recognized by ten best paper awards from IEEE and ACM, including the IEEE INFOCOM Test of Time Paper Award in 2023. He holds six U.S. patents. He authored/co-authored two graduate textbooks: *Applied Optimization Methods for Wireless Networks* (Cambridge University Press, 2014) and *Cognitive Radio Communications and Networks: Principles and Practices* (Academic Press/Elsevier, 2009). He was named an IEEE Fellow for contributions to the modeling and optimization of wireless networks. He was on the editorial boards of a number of IEEE and ACM transactions and journals. He served as the Steering Committee Chair of the IEEE INFOCOM conference. He was a member of the IEEE Communications Society Board of Governors. He was also a Distinguished Lecturer of the IEEE Communications Society.

Wenjing Lou (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Florida. She is currently a W. C. English Endowed Professor of computer science at Virginia Tech, Arlington, VA, USA. Her research interests include the cyber security field, with her current research interest focusing on wireless networks, privacy protection in machine learning systems, and security and privacy problems in the Internet of Things (IoT) systems. She is a fellow of ACM. She received the Virginia Tech Alumni Award for Research Excellence in 2018, which is the highest university-level Faculty Research Award. She received the INFOCOM Test-of-Time Paper Award in 2020. She served as the Steering Committee Chair for the IEEE CNS Conference from 2013 to 2020. She is currently a Steering Committee Member of IEEE INFOCOM. She was the TPC Chair for IEEE INFOCOM 2019 and ACM WiSec 2020. She served as the Program Director at U.S. National Science Foundation (NSF) from 2014 to 2017. She is a Highly Cited Researcher by the Web of Science Group.

Jeffrey H. Reed (Life Fellow, IEEE) is currently the Willis G. Worcester Professor with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA. He has co-authored more than 500 articles and books. In addition, he co-founded several commercial companies, including Federated Wireless, PFP Cybersecurity, and Cirrus360. He is also the Founding Director of Wireless@Virginia Tech, a university research center, and a co-founder of Virginia Tech's Hume Center for National Security and Technology, where he served as the Interim Director. He also served as the Interim Director of the Commonwealth Cyber Initiative and is currently the CTO. His research interests include wireless communications, cognitive radio, software radio, wireless security, telecommunications policy, and spectrum access. He is a fellow of the IEEE for contributions to software radio and communications signal processing and leadership in engineering education.

Luiz A. DaSilva (Fellow, IEEE) is currently the Bradley Professor of cybersecurity at Virginia Tech, Arlington, VA, USA. He serves as the Inaugural Executive Director of the Commonwealth Cyber Initiative, a consortium of 42 institutions of higher education in Virginia, USA, with a mission of research, innovation, and workforce development in cybersecurity. Previously, he held the Chair of Telecommunications at Trinity College Dublin, Ireland, where he led CONNECT, a research center in future networks involving ten universities in Ireland. He is a fellow of the IEEE for contributions to cognitive networks and wireless resource management. He has also been a fellow of Trinity College Dublin, a Distinguished Lecturer of the IEEE Communications Society, and a College of Engineering Faculty Fellow.